

Implementación de un sistema para el registro de eventos relacionados con rayos cósmicos.

Dora Luz Ballesteros Delgado

Trabajo de Grado para optar al título de Ingeniera Electrónica

Director

Carlos Andrés Angulo Julio

Magíster en Ing. Electrónica

Codirectores

Luis Alberto Núñez de Villavicencio Martínez

Ph.D. Física

Jesús Peña Rodríguez

Ph.D. (c) en Física

Universidad Industrial de Santander

Facultad de Ingenierías Fisicomecánicas

Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones

Bucaramanga

2020

Agradecimientos

Agradezco principalmente a Dios por permitirme alcanzar esta meta, seguidamente agradezco a mi familia por el apoyo que me brindaron económica y moralmente, a mi papá que se ha esforzado mucho por verme triunfar, a mi mamá por apoyarme y darme su voz de aliento cuando sentía que todo estaba perdido y finalmente a mis hermanos por creer siempre en mí.

Agradezco a Andrés Felipe Jaramillo por su apoyo en el diseño y ejecución de este proyecto, también le agradezco por escucharme, guiarme y brindarme su respaldo con tanto amor en esta etapa tan importante de mi vida.

Agradezco a mis amigos con quienes pasé grandes dificultades en esta etapa universitaria, gracias por estar conmigo en todo momento, por su apoyo y incondicionalidad.

Agradezco a mi director Carlos Angulo por el tiempo dedicado y por brindarme sus conocimientos que me ayudaron a alcanzar este logro. Agradezco a mi codirector Jesús Peña su dedicación incondicional en todo el proyecto, por su apoyo emocional y intelectual. También agradezco a mi codirector Luis Núñez por brindarme esta oportunidad, por creer en mí y por apoyarme en tantos inconvenientes presentados en el proyecto.

Tabla de Contenido

Introducción	15
1 Objetivos	17
2 Detección de rayos cósmicos	18
2.1 Rayos Cósmicos	18
2.2 Proyecto LAGO	19
2.2.1 Detectores WCD de LAGO	19
2.2.2 Tubos fotomultiplicadores	20
3 Sistema de adquisición del proyecto LAGO	23
3.1 Procesador de datos	25
3.1.1 Corrección de línea base	26
3.1.2 Regulación de polarización de los PMT	28
3.1.3 Discriminación de datos (TRIGGER)	30
3.2 Protocolo de comunicación	35
3.3 Hardware externo de emulación	37
3.4 Raspberry pi 3 B+	38
4 Implementación proyecto LAGO	40
4.1 FPGA	40
4.2 Sensores adicionales	43

4.2.1	GPS	43
4.2.2	Temperatura y presión atmosférica	44
4.3	Raspberry Pi B+	45
4.4	Integración de periféricos	46
5	Resultados	49
5.1	Comparación simulaciones de LAGO	50
5.1.1	Simulación Baseline versión original y versión actualizada	50
5.1.2	Simulación Rampa versión original y versión actualizada	53
5.1.3	Simulación Trigger versión original y versión actual	56
5.2	Salida de control para PMT's	58
5.3	Registro de datos	60
6	Trabajo futuro	64
7	Conclusiones	64
	Referencias	65

Lista de Figuras

Figura 1	Lluvia atmosférica extendida (EAS)	19
Figura 2	Ilustración de un Detector Cherenkov	20
Figura 3	Esquema de funcionamiento de un fotomultiplicador	21
Figura 4	Sistema de adquisición LAGO original	22
Figura 5	Componentes principales del sistema de adquisición	22
Figura 6	Esquema general del sistema de adquisición actualizado	24
Figura 7	Representación del sistema de adquisición	25
Figura 8	Diagrama de bloques del procesador de datos.	26
Figura 9	Circuito de actualización de baseline cada 2ms.	27
Figura 10	Circuito para acumular los datos provenientes de los ADC.	27
Figura 11	Circuito para acualizar el control de línea base cada 2 ms.	27
Figura 12	Salida del nivel de cuantización al DAC de la línea base.	28
Figura 13	Circuito para el registro del umbral	29
Figura 14	Circuito para modificar el ciclo útil de la PWM cada 2ms.	29
Figura 15	Circuito para generar la frecuencia de la PWM de (10 KHz).	29
Figura 16	Circuito de control del ciclo útil de la PWM	30
Figura 17	Circuito señal de control PWM, (OUT_REG) ciclo útil, (R_REG) frecuencia.	30
Figura 18	Máquina de estados del subsistema de activación	31
Figura 19	Circuito para el registro de los datos provenientes del ADC.	32

Figura 20	Circuito de umbral de activación	32
Figura 21	Circuito concatenación trama de datos a comunicar protocolo SPI	33
Figura 22	Circuito concatenación trama de datos a comunicar protocolo SPI	33
Figura 23	Máquina de estados del conteo de pulsos por segundo	34
Figura 24	Circuito de concatenación para la generación del PPS_FALSO.	34
Figura 25	Circuito para generar la señal PPS	35
Figura 26	Circuito de activación de PPS_FALSO_LED.	35
Figura 27	Esquema protocolo SPI	36
Figura 28	RTL protocolo de comunicación SPI	36
Figura 29	Diagrama de bloques hardware externo	38
Figura 30	Diagrama de bloques Raspberry Pi.	39
Figura 31	Icoboard con FPGA Lattice iCE40 de Trenz Electronic	40
Figura 32	Compatibilidad con la Raspberry Pi	41
Figura 33	Arquitectura iCE40	42
Figura 34	Recursos lógicos de la FPGA	43
Figura 35	GPS adafruit ultimate	44
Figura 36	Sensor de presión y temperatura HP03S	44
Figura 37	Características de salida del sensor de presión y temperatura	45
Figura 38	Diseño de la PCB	47
Figura 39	PCB de integración de periféricos	47
Figura 40	Implementación proyecto LAGO para pruebas con nexys 4-DDR	48
Figura 41	Implementación del hardware LAGO actualizado	48

Figura 42	Simulación del comportamiento de las señales en baseline, versión actualizada.	50
Figura 43	Simulación de las entradas de baseline, versión actualizada.	51
Figura 44	Simulación del comportamiento de las señales en baseline, versión original.	52
Figura 45	Simulación de las entradas de baseline, versión original.	53
Figura 46	Simulación rampa de polarización del proyecto LAGO, versión actualizada.	54
Figura 47	Ciclo útil de 10 ns para la PWM del proyecto LAGO, versión actualizada.	54
Figura 48	Actualización de la PWM del proyecto LAGO, versión original.	55
Figura 49	Ciclo útil de 20ns para la PWM del proyecto LAGO, versión original.	55
Figura 50	Ciclo útil de 20ns para la PWM del proyecto LAGO, versión original.	56
Figura 51	Señales de Trigger destacadas del proyecto LAGO, versión actualizada.	57
Figura 52	Señales de Trigger destacadas del proyecto LAGO, versión original.	58
Figura 53	Señal de control PWM, versión actualizada.	59
Figura 54	Señal de control PWM, versión actual.	59
Figura 55	Cabecera del archivo de datos del terminal de usuario proyecto LAGO.	60
Figura 56	Comparación de archivos de salida con datos de eventos considerados rayos cósmicos	61
Figura 57	Gráfica de evento hardware vs pulso interpolado	62
Figura 58	Registro de temperatura HP03S durante un registro de media hora.	63
Figura 59	Registro de presión HP03S durante un tiempo de registro de media hora.	63

Lista de Tablas

Tabla 1	Mapa de memoria protocolo SPI.	37
Tabla 2	Características Raspberry Pi3 Model B+	45

Glosario

EAS *Extensive Air Showers* : Lluvias atmosféricas extendidas o cascada de partículas son partículas secundarias procedentes del espacio. Estas se originan por la interacción de los núcleos de los átomos y compuestos de los gases que constituyen la atmósfera con los rayos cósmicos primarios.

FPGA *Field Programmable Gate Array*: Matriz de puertas lógicas programable del inglés field-programmable gate array. Es un circuito integrado cuya interconexión y funcionalidad puede ser configurada por medio de lenguaje de descripción de hardware (HDL).

LAGO *The Latin American Giant Observatory*: observatorio de astropartículas compuesto por detectores Cherenkov en agua ubicados en nueve países de América Latina (Argentina, Bolivia, Brasil, Colombia, Ecuador, Guatemala, México, Perú y Venezuela) y España. Los objetivos son: estudio del componente de alta energía de las explosiones de rayos gamma a gran altura, analizar y comprender el clima espacial mediante la modulación solar de los rayos cósmicos galácticos y medición de la radiación a nivel del suelo.

PMT *Photo-Multiplier Tubes* (Tubos fotomultiplicadores): Detectores de luz extremadamente sensibles diseñados para reaccionar ante una mínima cantidad de luz incidente. Operan con una fuente de alto voltaje conectada a una red resistiva que polariza sus terminales.

Radiación Cherenkov Es la luz emitida por un medio transparente cuando partículas cargadas

lo cruzan a una velocidad mayor a la de la luz en ese medio. La luz se emite en dirección al movimiento de la partícula.

WCD *Water Cherenkov Detector* (Detector Cherenkov de Agua): Sistema de detección que consiste en un tanque de agua sellado para evitar el ingreso de luz desde el exterior y recubierto en su interior con un material que permite una alta difusión lumínica. Su funcionamiento se basa en el efecto Cherenkov según el cual cuando una partícula cargada rompe la velocidad de la luz en el medio (agua) genera un cono luminoso de cuya intensidad se puede obtener información del evento.

Resumen

Título: Implementación de un sistema para el registro de eventos relacionados con rayos cósmicos *

Autor: Dora Luz Ballesteros Delgado **

Palabras Clave: WCD, PMT, Radiación Cherenkov, LAGO, FPGA, EAS.

Descripción: Este proyecto tiene como objetivo mejorar el sistema de adquisición de datos usado en la red de colaboración LAGO (*Latin American Giant Observatory*), la cual implementa un sistema digital para la detección de eventos que pueden ser considerados fenómenos debidos a rayos cósmicos. Con esta mejora se busca darle continuidad al proyecto LAGO así como independencia del sistema de adquisición respecto a las referencias de FPGA que se encuentran en el mercado ya que en el ciclo completo de diseño se han empleado herramientas de software libre.

El sistema de preprocesamiento de señales se rediseñó para que fuese llevado a cabo por medio de un procesador de propósito específico implementado en la FPGA. También se implementó un conjunto de bloques jerárquicos tales como: control de voltaje de polarización para los tubos fotomultiplicadores, corrección de línea base y discriminación de datos. Esto permite aumentar en 10 MHz la frecuencia de muestreo y mejorar la resolución de la señal de entrada hasta 4 veces. Adicionalmente, el sistema de adquisición integra periféricos para establecer el tiempo de adquisición, la ubicación geográfica, la presión atmosférica y la temperatura ambiente por medio de una Raspberry Pi que toma lecturas de un GPS y de sensores de temperatura y presión. Así mismo, esta Raspberry Pi se comunica a la FPGA por protocolo SPI para poder generar un reporte y guardar en un archivo la información de las señales registradas.

* Trabajo de grado

** Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones.
Director: Carlos Andrés Angulo Julio, Magíster en Ing. Electrónica.

Abstract

Title: Implementation of a data acquisition system for cosmic rays. *

Author: Dora Luz Ballesteros Delgado **

Keywords: WCD, PMT, Cherenkov Radiation, LAGO, FPGA, EAS.

Description: This project aims to improve the data acquisition system used in the LAGO (Latin American Giant Observatory) collaboration network, which implements a digital system for the detection of events that could be considered phenomena due to cosmic rays. This improvement seeks to give continuity to the LAGO project as well as independence of the acquisition system regarding FPGA references that are on the market due to free software tools have been used in the complete design cycle. The signal preprocessing system was redesigned to be carried out by a specific purpose processor implemented in the FPGA. A set of hierarchical blocks such as bias voltage control for photomultiplier tubes, baseline correction, and data discrimination were also implemented in the FPGA. This allows the sampling rate to be increased by 10 MHz and the resolution of the input signal to be improved up to 4 times. Moreover, the acquisition system integrates peripherals to establish the acquisition time, geographic location, atmospheric pressure, and ambient temperature using a Raspberry Pi that takes readings from a GPS and from temperature and pressure sensors. Likewise, this Raspberry Pi communicates with the FPGA by SPI protocol in order to generate a report and store in a file the information of the registered signals. This file can be sent to another device in order to analyze the data of the events that can be considered cosmic rays.

* Bachelor Thesis

** Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones. Director: Carlos Andrés Angulo Julio, Magíster en Ing. Electrónica.

Introducción

Los rayos cósmicos son partículas elementales y fragmentos atómicos provenientes del espacio que interactúan con la atmósfera terrestre, con un rango de energía desde 10^9 eV hasta 10^{20} eV que viajan casi a la velocidad de la luz. Estos contribuyen a la radiación natural y son uno de los fenómenos más interesantes y desconocidos en la física de partículas y astropartículas. (Supanitsky, 2007).

Cuando un rayo cósmico de alta energía interactúa con los núcleos atómicos de la atmósfera terrestre desencadena reacciones nucleares que originan nuevas partículas secundarias que se multiplican en un proceso de cascada. (Estupiñán López et al., 2015).

Para la detección de rayos cósmicos se pueden utilizar varios métodos tales como detectores ubicados a nivel del suelo como: WCDs, telescopios de fluorescencia y detectores de centelleo. Estos métodos tienen como fin caracterizar la energía y dirección de incidencia de las partículas.

LAGO es un observatorio de rayos cósmicos extendido compuesto por detectores Cherenkov de agua (WCD) situados en diferentes países de América Latina. El objetivo de dicho proyecto es crear una red de detección de rayos cósmicos con miras a realizar análisis de eventos de clima espacial. Cada punto en la red de detección consta de WCDs, sistemas para la adquisición de datos, sensores que generan información relevante para el post-procesamiento de las señales y dispositivos de procesamiento y almacenamiento de la información.

Actualmente existen tres WCDs dispuestos dentro de la Universidad Industrial de Santander

(UIS) que forman parte del proyecto LAGO. El arreglo de WCDs tiene forma de triángulo cuasi equilátero con el fin de facilitar el cálculo de los parámetros para la dirección de la EAS mediante la radiación depositada por los rayos cósmicos secundarios al atravesar el tanque detector (Hernandez Barajas et al., 2018).

En este proyecto se presenta una actualización del sistema de registro de eventos relacionados con rayos cósmicos secundarios para el proyecto LAGO. El diseño utilizado es una derivación del hardware actual. Este libro está organizado de la siguiente manera: en el capítulo 1 se exponen los objetivos planteados para el proyecto. El Capítulo 2 contiene los conceptos generales para el desarrollo del proyecto. En el Capítulo 3 se describe el sistema de adquisición LAGO mientras que en el Capítulo 4 se presenta la implementación del proyecto. En el Capítulo 5 se muestran los resultados obtenidos y se finaliza con el trabajo futuro y las conclusiones en el Capítulo 6 y el capítulo 7 respectivamente.

1. Objetivos

Objetivo general

Implementar un sistema digital para el registro de eventos relacionados con rayos cósmicos secundarios obtenidos por medio de los Detectores Cherenkov de Agua (por sus siglas en inglés WCD) del arreglo Guane.

Objetivos específicos

Implementar en FPGA circuitos a nivel RTL para lograr la regulación de la fuente de alto voltaje para los PMT del detector, la corrección de línea base y la discriminación de señal.

Transmitir los datos capturados desde el FPGA hacia la raspberry.

Comunicar la raspberry con dos periféricos que permitan obtener información relacionada con el entorno del WCD (ubicación, hora, temperatura ambiente y presión atmosférica).

Registrar en raspberry la información relacionada con aquellos eventos que puedan ser considerados debidos a rayos cósmicos.

2. Detección de rayos cósmicos

2.1. Rayos Cósmicos

Los rayos cósmicos primarios son partículas procedentes del espacio cuya energía se debe a su gran velocidad. Éstas son principalmente núcleos atómicos, fotones y partículas neutras.

Cuando los rayos cósmicos primarios interactúan con la atmósfera terrestre desencadenan un proceso estocástico conocido como lluvia atmosférica extendida (EAS, por sus siglas en inglés). Esta consiste en una cascada de partículas secundarias (radiación electromagnética de muones y nucleones) que se dirigen hacia la superficie de la Tierra a altas velocidades en la dirección del rayo cósmico incidente (Asorey, 2012). Esta cascada es producida por la interacción de un rayo cósmico primario con átomos de la parte superior de la atmósfera.

Actualmente se utilizan dos tipos de métodos para la detección de rayos cósmicos: los métodos indirectos y los directos. En los métodos directos las partículas primarias inciden directamente con el detector, por lo cual los dispositivos de detección se encuentran situados en satélites, globos o aviones. Los métodos indirectos detectan la cascada de partículas secundarias producida por la interacción con la atmósfera. Para la detección de éstas se utilizan detectores ubicados en tierra como WCD, telescopios de fluorescencia y detectores de centelleo (Schlaepfer, 2003). En la Figura 1 se observa como las EAS pueden ser detectadas a nivel del suelo mediante arreglos de detectores Cherenkov de agua.

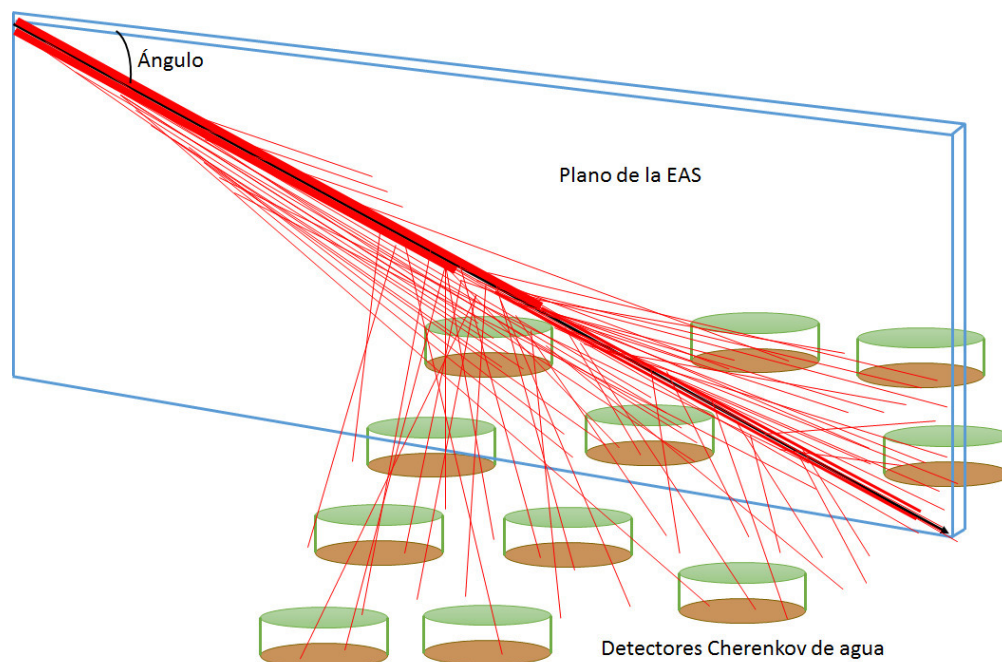


Figura 1. Lluvia atmosférica extendida (EAS) detectada mediante un arreglo de varios WCD. Procesando el tiempo con el cual el evento llega a cada detector de superficie se determina el ángulo de incidencia de las partículas. (Hernandez Barajas et al., 2018).

2.2. Proyecto LAGO

El proyecto LAGO (por sus siglas en inglés, *The Latin American Giant Observatory*) es un observatorio de astropartículas que se dedica al estudio de temas relacionados con astrofísica y clima espacial y que cuenta con equipos instalados en diferentes países de América Latina.

2.2.1. Detectores WCD de LAGO. El proyecto LAGO utiliza tanques de agua como detectores de partículas. Un WCD típico de LAGO es un tanque lleno de agua purificada cuyo volumen oscila entre 1 m^3 y 40 m^3 . En su interior tiene una bolsa hecha por un tejido altamente difusivo y reflectante de Tyvek ®1073D para contener todo el volumen de agua y aumentar la eficiencia de detección. El objetivo de este recubrimiento es difundir los fotones Cherenkov captados por

los tubos fotomultiplicadores (PMT, por sus siglas en inglés) y reducir la dependencia de la señal de la trayectoria de las partículas secundarias dentro del detector, ya que con esto los fotones se distribuirán uniformemente dentro del tanque (Ver Figura 2).

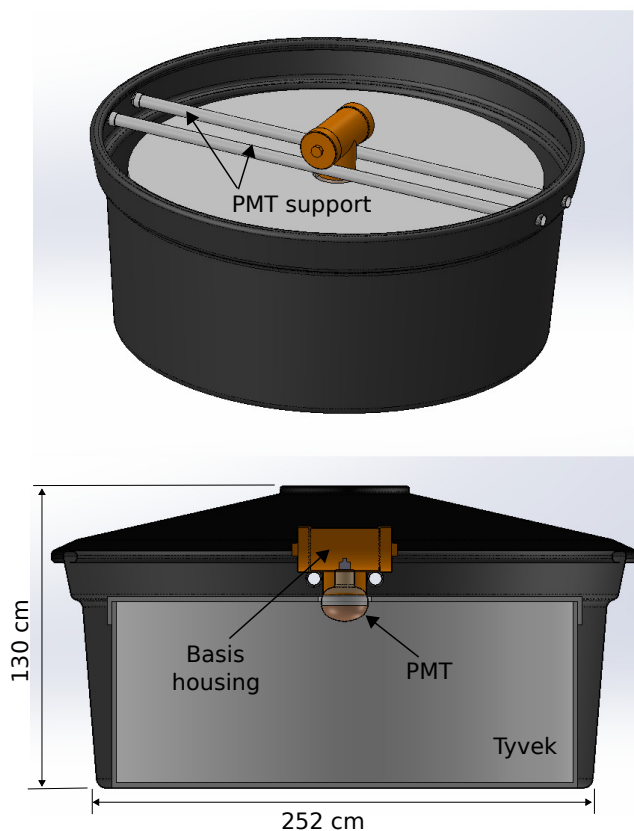


Figura 2. Ilustración de un Detector Cherenkov. (Hernandez Barajas et al., 2018)

2.2.2. Tubos fotomultiplicadores. Los PMT son los dispositivos fotosensores más empleados debido tanto a su gran versatilidad como sus características (respuesta rápida, alta sensibilidad y alto factor de ganancia). Un tubo fotomultiplicador funciona como transductor y a su vez como amplificador, es decir, que a partir de una luz detectada se produce una corriente eléctrica fácilmente medible.

En la Figura 3 se muestra el funcionamiento interno de un PMT. Cuando los fotones de luz

visible alcanzan el fotocátodo, éste emite fotoelectrones de baja energía. Estos electrones son acelerados y multiplicados en campos eléctricos secuenciales aplicados entre los electrodos del PMT llamados dínodos. En los dínodos la señal eléctrica es suficientemente grande para poder ser manejada con amplificadores y analizadores de pulsos convencionales. (Kaptanoglu, 2018).

LAGO utiliza tubos fotomultiplicadores PMT Hamamatsu R5912 de 8 pulgadas desarrollados por Hamamatsu Photonics los cuales se encuentran ubicados sobre el volumen de agua donde inciden los fotones de Cherenkov. Estos PMT tienen 10 etapas enfocadas linealmente y su circuito de alimentación está constituido por un divisor de voltaje, una fuente de alimentación de alto voltaje y un preamplificador para acondicionar las amplitudes de los pulsos de salida. La tensión de polarización del PMT está en el rango de 0 V a 2000 V respecto a tierra y puede controlarse con una baja tensión en el rango de 0 V a 5 V.

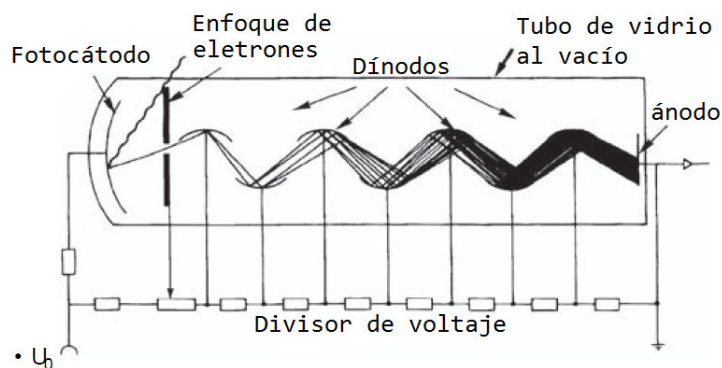


Figura 3. Esquema de funcionamiento de un fotomultiplicador. El fotocátodo convierte la energía de luz incidente en una corriente de electrones por efecto fotoeléctrico. El campo eléctrico creado entre los dínodos permite acelerar y guiar los electrones a lo largo del multiplicador. El divisor de voltaje está compuesto por un arreglo de resistencias que dividen el voltaje y alimentan los dínodos. (Hernandez Barajas et al., 2018)

El sistema de adquisición implementado por LAGO permite suministrar las tensiones requeridas en la base de PMT. Un detector de WCD normalmente emite pulsos con un tiempo de subida de

~ 10 ns y un tiempo de bajada de ~ 70 ns. La forma de los pulsos que vienen del detector dependen de la pureza del agua, la reflectividad del material de difusión y la geometría del tanque. (Haro et al., 2016). En la Figura 4 se muestra el sistema de adquisición implementado en uno de los detectores instalado en la UIS, mientras que en la Figura 5 se muestra el diagrama de bloques de un detector WCD de LAGO.

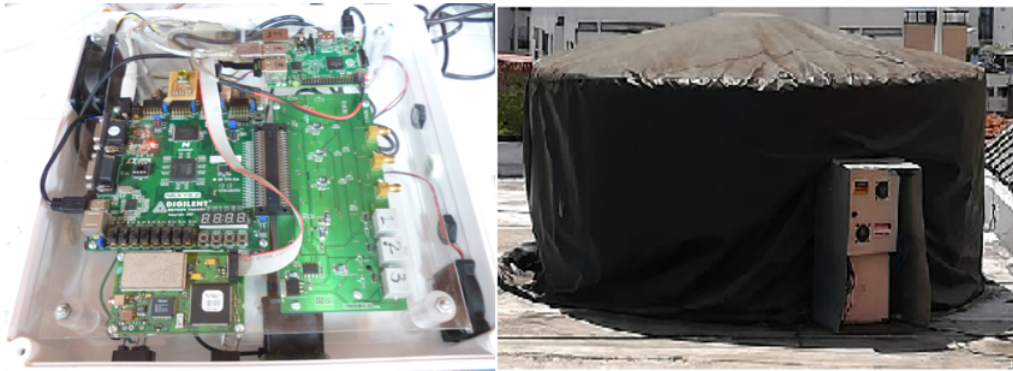


Figura 4. Sistema de adquisición LAGO original. A la derecha: WCD Chitagá ubicado el campus central de la Universidad Industrial de Santander en Bucaramanga, Colombia. A la izquierda: electrónica del sistema de adquisición del detector.

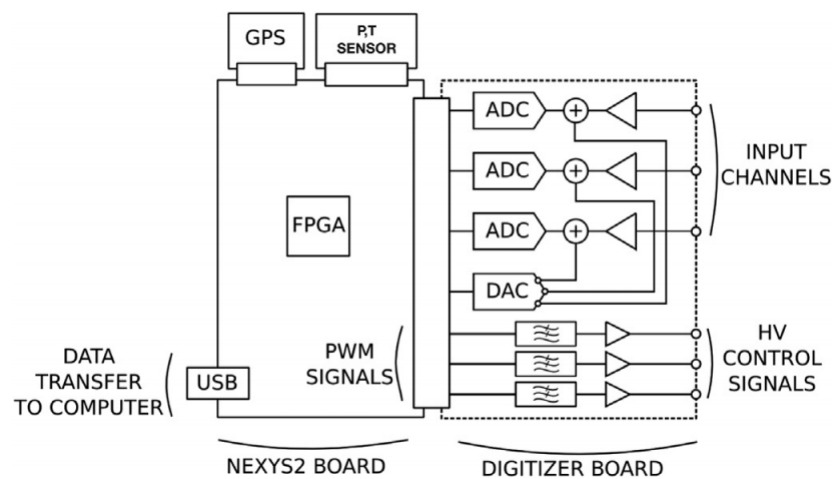


Figura 5. Componentes principales del sistema de adquisición: tarjeta digitalizadora de tres canales, placa Nexys-II y periféricos (GPS y sensores de temperatura y presión). (Haro et al., 2016).

3. Sistema de adquisición del proyecto LAGO

El proyecto LAGO esta compuesto por detectores de agua Cherenkov ubicados en toda América latina. Estos detectores cuentan con un sistema de adquisición electrónico que se compone de una etapa analógica para adquisición y una digital para el procesamiento de eventos relacionados con rayos cósmicos. El sistema de adquisición del proyecto LAGO se muestra en la Figura 5 el cual esta conformado por una tarjeta electrónica que digitaliza pulsos de tres canales independientes a una frecuencia de 40 MHz, cada canal de adquisición tiene un sumador destinado al control de linea base que se genera por las variaciones de temperatura, la tarjeta Nexys 2 es la responsable del procesamiento de datos, ajuste de umbrales y ajuste de voltajes. Dos de los dispositivos que conforman este sistema se encuentran fuera del mercado por tanto se pone en riesgo la continuidad del proyecto. Por esta razón en este proyecto se busca actualizar la etapa de procesamiento digital haciendo uso de herramientas libres que den independencia respecto a las FPGA en el proyecto.

En esta sección se describe la metodología usada para alcanzar cada uno de los objetivos planteados en el plan del proyecto. Se implementa a nivel HDL los circuitos diseñados para discriminar, digitalizar y registrar aquellos pulsos que se consideran eventos. Además se registra la hora, geolocalización del detector, temperatura y presión atmosférica. Los datos son agrupados en un archivo de salida que se entrega para su posterior análisis.

El sistema de adquisición tiene 3 canales, para lo cual se emplea el ADC AD9235 que tiene frecuencia de muestreo máxima de 50 MHz y 12 bits de resolución.

En la Figura 6 se observa el sistema de adquisición de LAGO el cual se compone de los siguientes dispositivos:

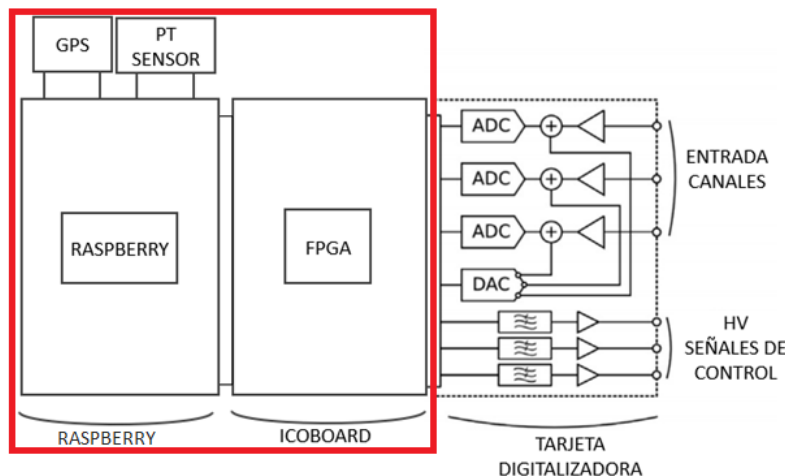


Figura 6. Esquema general del sistema de adquisición actualizado, resaltando en rojo las etapas abordadas de este proyecto.

- Tarjeta electrónica que digitaliza pulsos de 3 canales independientes con salida paralela de 12 bits y una frecuencia de 50 MHz,
- Tarjeta Icoboard con FPGA compatible con herramientas de desarrollo libre : En ella se implementa un conjunto de bloques jerárquicos que permiten detectar eventos relacionados con rayos cósmicos
- Raspberry Pi3 model B+: Permite la comunicación con los periféricos y a su vez genera y guarda un archivo con información sobre las señales captadas y los sensores externos con el fin de establecer el tiempo de adquisición, la ubicación geográfica, la presión atmosférica y la temperatura.
- Dos periféricos que permitan obtener información relacionada con el entorno del WCD (ubicación, hora, temperatura ambiente y presión atmosférica).

Nota: la información proveniente de la tarjeta de digitalización será simulada con otra FPGA desde archivos existentes.

En la Figura 7 se muestra un diagrama de bloques del sistema de adquisición implementado. Este está compuesto por: FPGA (Icoboard), Raspberry Pi 3 (archivo de salida), hardware externo (tarjeta de digitalización).

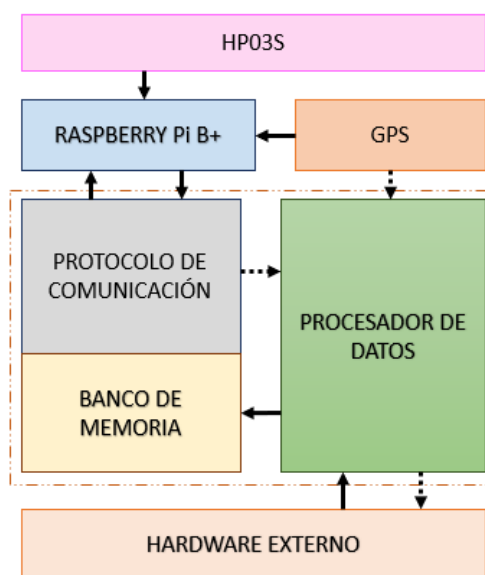


Figura 7. Representación del sistema de adquisición. Flecha discontinua: Señal de control, flecha continua: Señal de control, línea discontinua naranja: FPGA.

3.1. Procesador de datos

En la Figura 8 se hace una descripción funcional mediante diagrama de bloques de los circuitos de control del PMT (RAMPA), detección señal (TRIGGER), corrección de línea base (BASELINE).

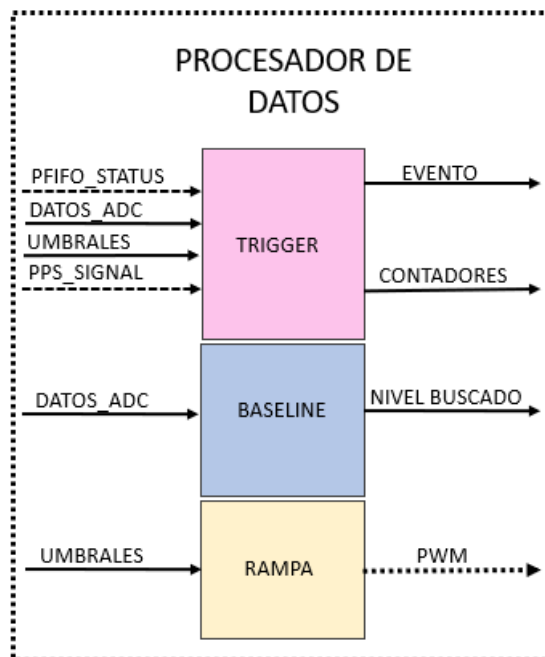


Figura 8. Diagrama de bloques del procesador de datos.

3.1.1. Corrección de línea base. Esta corrección se realiza con el fin de estabilizar la electrónica de adquisición frente a los efectos ambientales como la temperatura. Es decir, si la temperatura aumenta esto provocará que la línea de base se incremente, y viceversa. Dado que el sistema implementado compara el voltaje instantáneo en el canal ADC con una tensión de referencia (umbral de detección), cualquier cambio en el nivel de línea base podría afectar las mediciones. Para evitar esta afectación y también el ruido electrónico se cuenta con circuitos HDL que logran mantener un valor estable de línea base en 50 niveles de un convertidor analógico a digital (ADC) ($\sim 49\text{mV}$). El voltaje de línea base controlado por la FPGA se aplica a través de un convertidor de digital a analógico (DAC) de 12 bits el cual se actualiza cada 2 ms. A continuación se hace una explicación de cada uno de los bloques implementados en la FPGA para la corrección de línea base.

Para lograr una actualización de 2 ms se diseña el circuito mostrado en la Figura 9 donde *refresh*

rate toma un valor de 100000, este valor se establece teniendo en cuenta la frecuencia del reloj de 50MHz, (período de 20 ns), entonces:

$$\#ciclos \times periodo = tiempo \rightarrow 100000 \times 20ns = 2ms \quad (1)$$

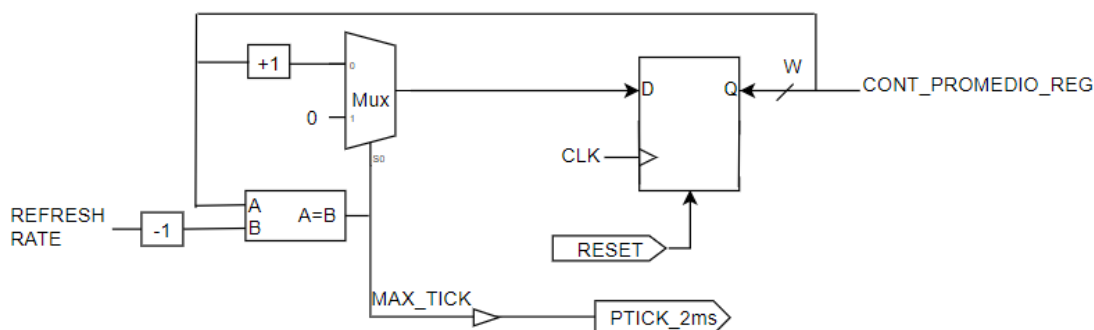


Figura 9. Circuito de actualización de baseline cada 2ms.

El circuito de la Figura 10 muestra como los datos provenientes del ADC se acumulan durante 2 ms (*MAX_TICK*) para luego hacer una comparación y lograr estabilizar la señal en ~ 49 mV como se establece en (Haro et al., 2016). El nivel de cuantización suministrado al DAC (MAX5501) se realiza en el circuito de la Figura 11 donde se analiza la salida del acumulado (*ADC_PROM_REG*) en el contador de los ADC.

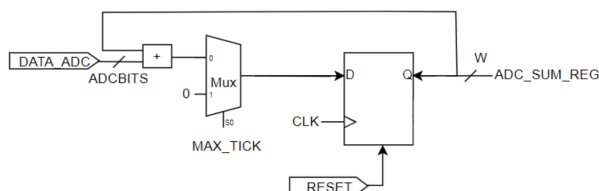


Figura 10. Circuito para acumular los datos provenientes de los ADC.

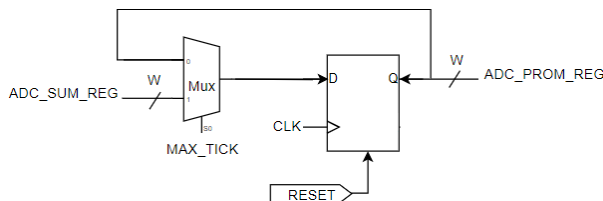


Figura 11. Circuito para actualizar el control de línea base cada 2 ms.

En (Haro et al., 2016) se diseña un circuito inversor, cuya entrada es una señal analógica invertida. Es decir, para aumentar la salida se debe disminuir la entrada y viceversa. En el circuito

mostrado en la Figura 12 es la señal que llega al multiplexor (MUX 2-1) y se actualiza cada 2 ms. Para el cálculo del módulo del circuito se tienen en cuenta los ciclos de reloj necesarios para generar la señal periódica de 2 ms. En este caso, el nivel de voltaje buscado coincide con 50 niveles de cuantización, este valor se basa en la teoría implementada en la versión original del proyecto, como se observa en la ecuación (2)

$$\#ciclos \times nivel_de_cuantización = umbral \rightarrow 100000 \times 50 = 5000000 \quad (2)$$

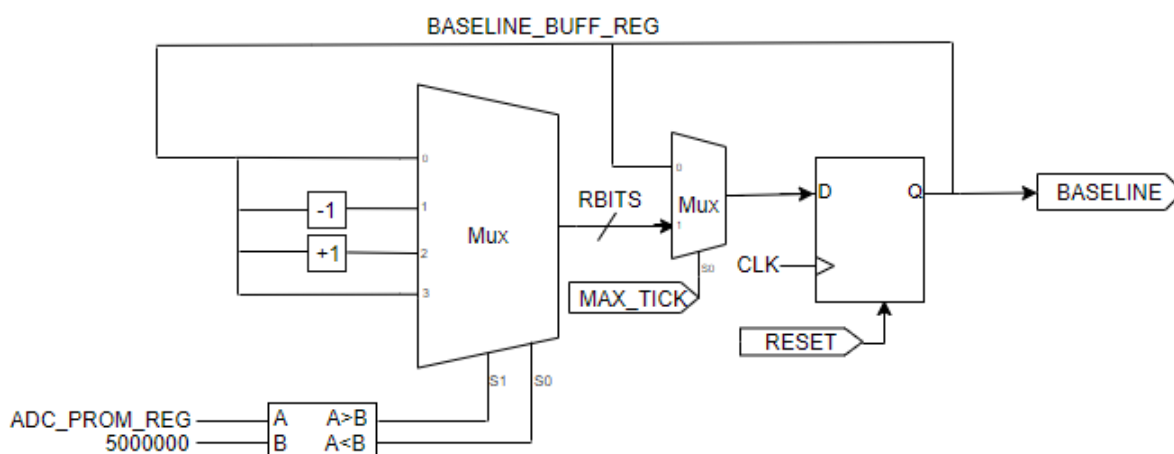


Figura 12. Salida del nivel de cuantización al DAC de la línea base.

3.1.2. Regulación de polarización de los PMT. Para el control de la fuente de alto voltaje que polariza el PMT se genera una rampa para aumentar y/o disminuir el voltaje. La rampa protege los PMT de cambios bruscos de voltaje que puedan dañarlo. El usuario puede programar un set point de tensión (DATA_IN) en el rango de 0 a 1023 en la terminal, lo que se traduce a un voltaje de 0 a 2000 V en el circuito de regulación de la fuente de polarización, como lo muestra el circuito de la Figura 13.

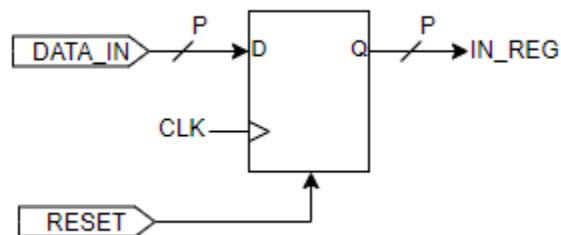


Figura 13. Circuito para el registro del umbral

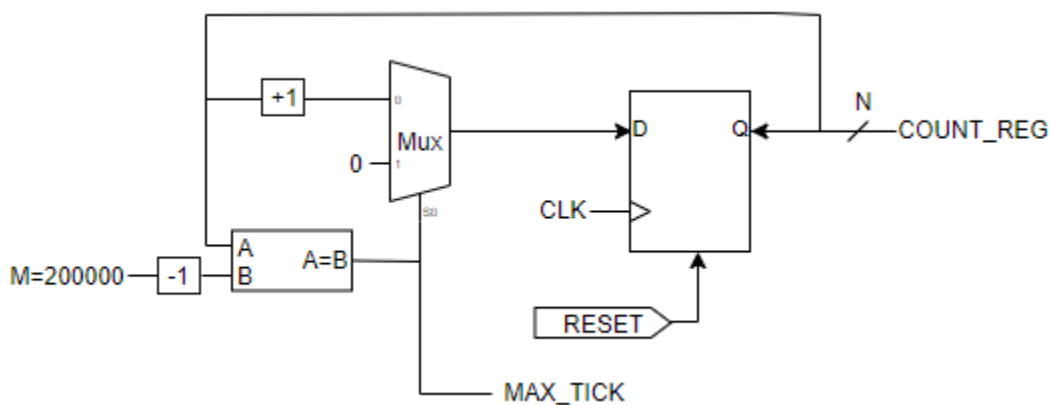


Figura 14. Circuito para modificar el ciclo útil de la PWM cada 2ms.

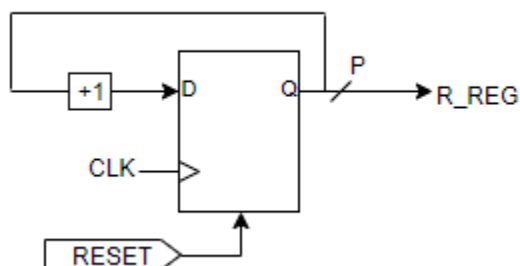


Figura 15. Circuito para generar la frecuencia de la PWM de (10 KHz).

El control de la PWM se establece con una frecuencia de 10 KHz Figura 15 y una actualización cada 2ms Figura 14 para lograr el control de su ciclo útil mediante la comparación respecto al umbral establecido por el usuario Figura 16.

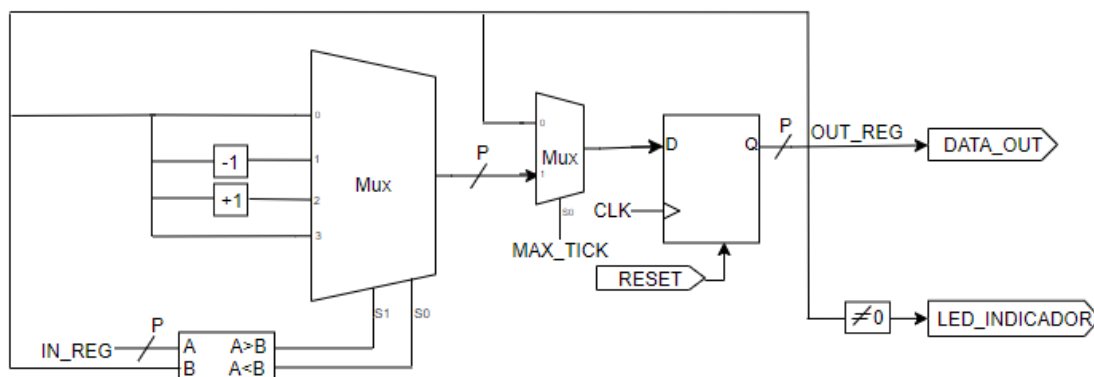


Figura 16. Circuito de control del ciclo útil de la PWM. (IN_REG) es el umbral establecido por el usuario, (MAX_TICK) actualización cada 2 ms, (OUT_REG) es el porcentaje de ciclo útil.

En el circuito de la Figura 17 se calcula la PWM a partir de los parámetros generados anteriormente. (OUT_REG) establece el porcentaje del ciclo útil y (R_REG) la frecuencia. De esta manera se ajusta el voltaje del PMT por medio de un conversor que interpreta la señal PWM y genera un voltaje de salida en el rango 0 a 2000V.

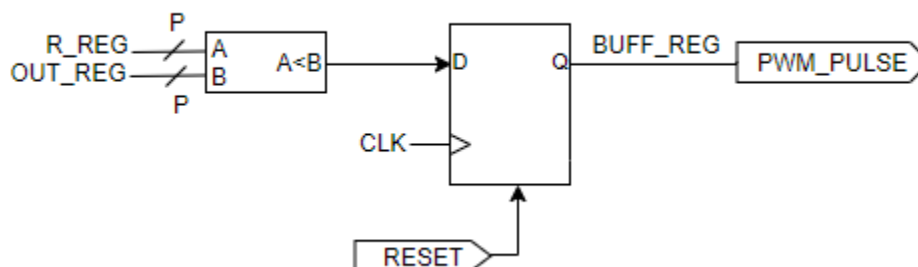


Figura 17. Circuito señal de control PWM, (OUT_REG) ciclo útil, (R_REG) frecuencia.

3.1.3. Discriminación de datos (TRIGGER). La discriminación de los datos determina que pulsos son los que deben almacenarse. La condición $V_i > V_{thr}$ determina que señal se considera un evento, donde V_i es la señal de entrada (DATA_ADC) y V_{thr} es el umbral (TRIGG_SET). Según los análisis hechos . (Haro et al., 2016) un pulso completo se puede adquirir en una ventana

de tiempo de 300 ns, garantizando que no se pierda información del evento adquirido. En este proyecto, son 15 muestras teniendo en cuenta la frecuencia de muestreo de 50 MHz. Todos los pulsos adquiridos se catalogan así: el conteo de pulsos registrados, el tiempo de ocurrencia y una etiqueta de 3 bits que identifica cual de los canales cumplió la condición de disparo.

La Figura 18 muestra una máquina de estados que controla todo el sistema de activación, esta permite controlar la información que se debe almacenar en orden secuencial en cada evento.

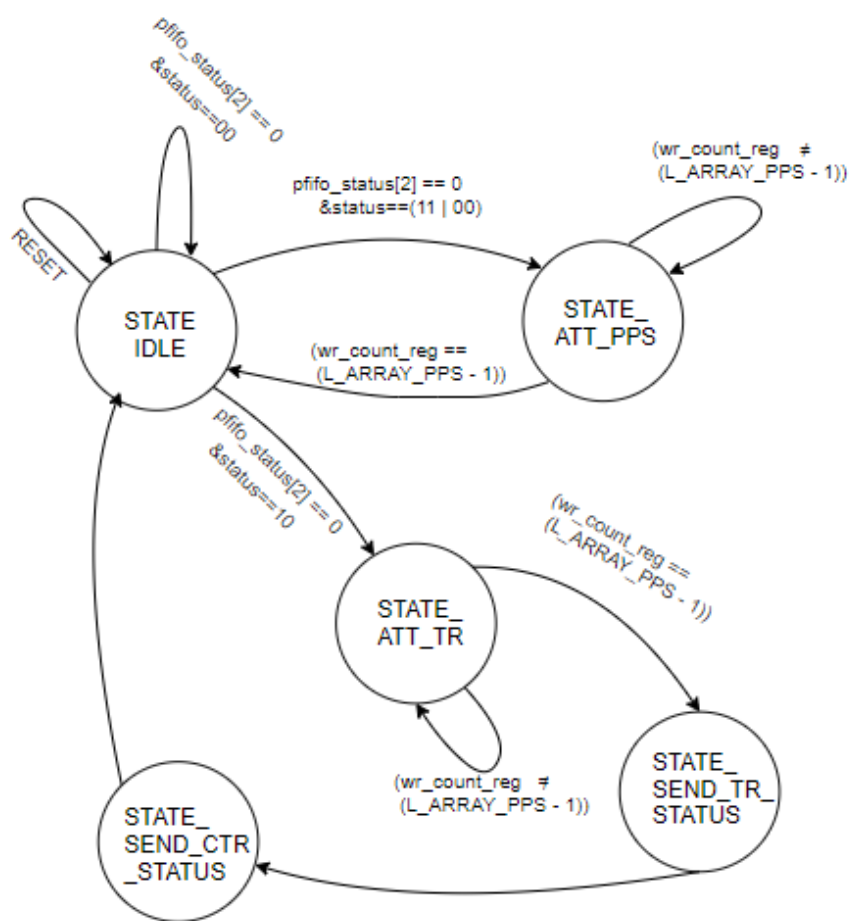


Figura 18. Máquina de estados del subsistema de activación, (STATE_IDLE) espera, (STATE_ATT_PPS) reinicio de contadores cada segundo, (STATE_ATT_PPS) muestreo del evento, (STATE_SEND_TR_STATUS) contador del tiempo de ocurrencia del evento, (STATE_SEND_CTR_STATUS) contador de eventos.

El registro de desplazamiento Figura 19 se encarga de entregar los datos provenientes de los

ADC. En el registro [15] se ingresa el dato y se desplaza hasta el registro [0], esto permite hacer una comparación con el umbral de activación y almacenar de manera simultánea la información de los 15 registros de cada canal

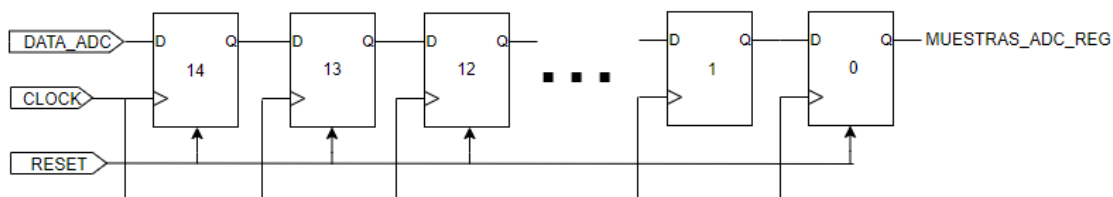


Figura 19. Circuito para el registro de los datos provenientes del ADC.

En la Figura 20 se tiene un conjunto de comparadores conectados al registro [3, 2, 1] que determinan si el registro [3] es mayor y los registros [2,1] son menores al umbral de activación se procede a almacenar todas las posiciones del registro anterior de la siguiente manera: se concatena la información almacenada en los registros de los 3 canales y se envía la trama resultante por comunicación SPI de modo que pasados 15 ciclos de reloj se ha comunicado todas las muestras del evento. Ver Figura 20

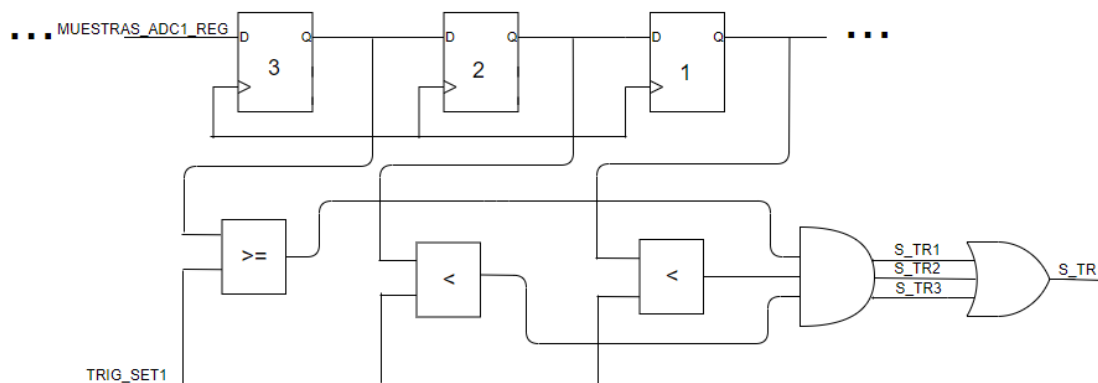


Figura 20. Circuito de umbral de activación se compara la señal de entrada digitalizada (MUESTRAS_ADC_REG) con el umbral de activación establecido por el usuario (TRIGG_SET), si se cumple la condición de disparo se activa la señal (S_TR)

El circuito de la Figura 21 concatena una bandera '10' con el contador de eventos (CONT_TRIGG_REG) el cual registra el conteo de los eventos cada vez que se cumple la condición de disparo en alguno de los 3 canales.

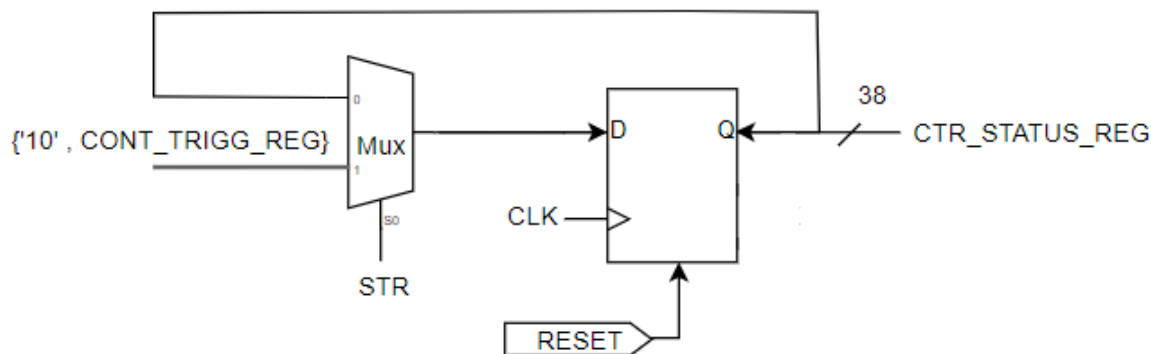


Figura 21. Circuito concatenación trama de datos a comunicar (CTR_STATUS_REG), (S_TR) habilitador del disparo, (CONT_TRIGG_REG) contador de eventos.

El circuito mostrado en la Figura 22 se activa cada vez que se cumple la condición de disparo. Este registra el tiempo de ocurrencia, y asigna una máscara de 3 bits [S_TR3, S_TR2, S_TR1] para detectar cual canal se dispara .

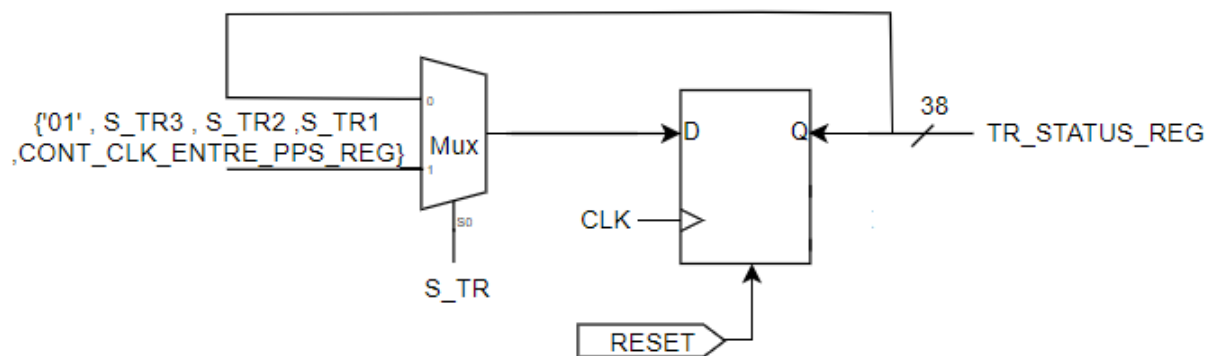


Figura 22. Circuito de concatenación de la trama de los datos a comunicar (TR_STATUS_REG), etiquetas de activación para cada canal (S_TR3,S_TR2,S_TR1), contador tiempo entre cada evento (CONT_CLK_ENTRE_PPS_REG).

Para controlar el reinicio sincronizado de los contadores de tiempo de ocurrencia se diseña la

siguiente máquina de estados que dispone de la señal pulso por segundo (PPS) la cual proviene del GPS. Ver Figura 23

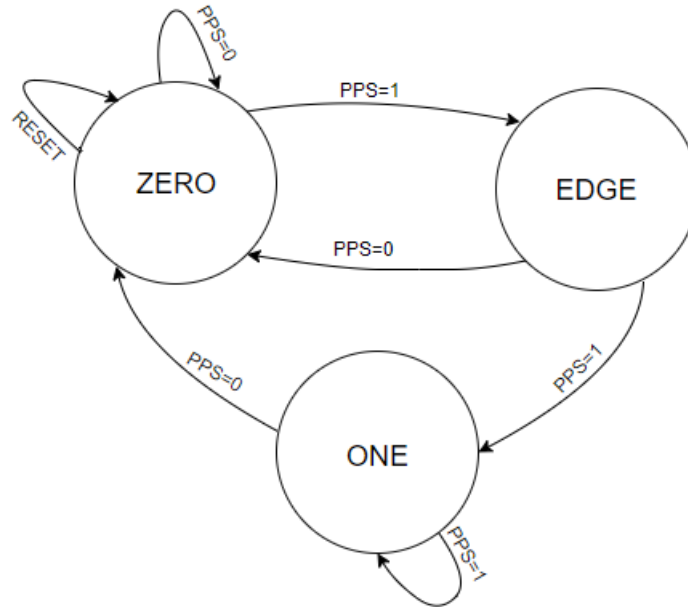


Figura 23. Máquina de estados del conteo de pulsos por segundo, (ZERO) estado de reposo, (PPS) señal del GPS que permite activar el estado (EDGE) y por consiguiente el estado (ONE).

En caso de no disponer de GPS, un circuito auxiliar divisor de frecuencia a 1 Hz se encarga de contar los flancos ascendentes de la señal de reloj clk de la FPGA para simular la señal PPS y generar un PPS_FALSO, con el fin de no afectar la adquisición de los eventos. (Ver Figura 24).

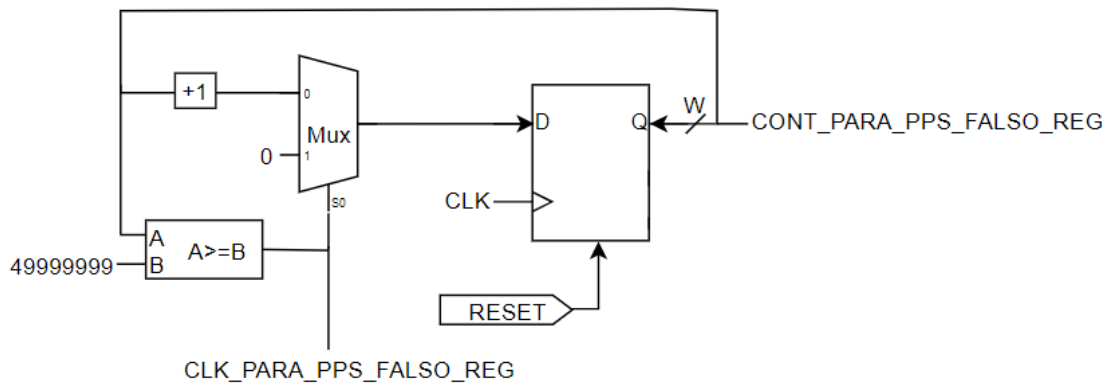


Figura 24. Circuito de concatenación para la generación del PPS_FALSO.

El circuito de la Figura 25 modula el ciclo útil del PPS_FALSO a 100 ms.

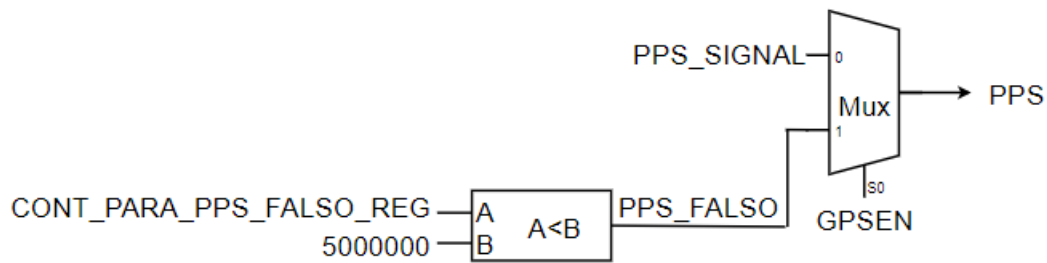


Figura 25. Circuito para generar la señal PPS, selecciona la señal proveniente del GPS (PPS_SIGNAL) o una señal simulada (PPS_FALSO), en caso de ausencia del GPS (GPSEN).

Si se activa el PPS_FALSO se activa un led de indicación se muestra en la Figura 26

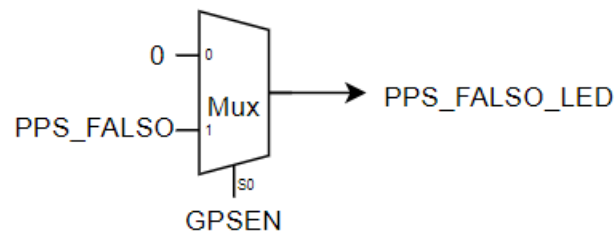


Figura 26. Circuito de activación de PPS_FALSO_LED.

3.2. Protocolo de comunicación

Este bloque logra la comunicación entre la FPGA y la Raspberry para la transmisión y recepción de la información; el protocolo se basa en el estándar SPI, donde la Raspberry es el maestro y la FPGA el esclavo. Este protocolo fue elegido teniendo en cuenta que la tasa de detección de un WCD típico en LAGO puede alcanzar valores hasta de 2 KHz. (Hernandez Barajas et al., 2018)

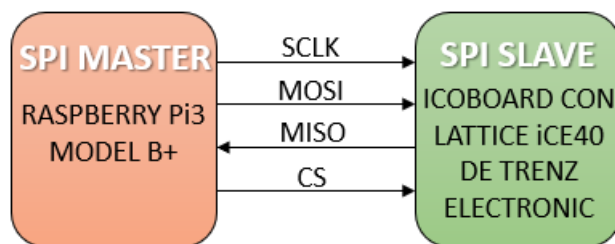


Figura 27. Protocolo SPI : SCLK señal de reloj proveniente del maestro Raspberry Pi 3, MOSI señal que permite llevar los bits del maestro hacia el esclavo, MISO señal que permite llevar los bits del esclavo al maestro y CS señal que se encarga de habilitar el esclavo.

El protocolo SPI utilizado en el proyecto fue basado en (Charkster, 2020). En el cual se implementa un circuito SPI Slave y un mapa de memoria diseñado para la comunicación entre una FPGA y una Raspberry. En la Figura 28 se muestra el diagrama RTL (*Register-transfer-level*) del protocolo implementado.

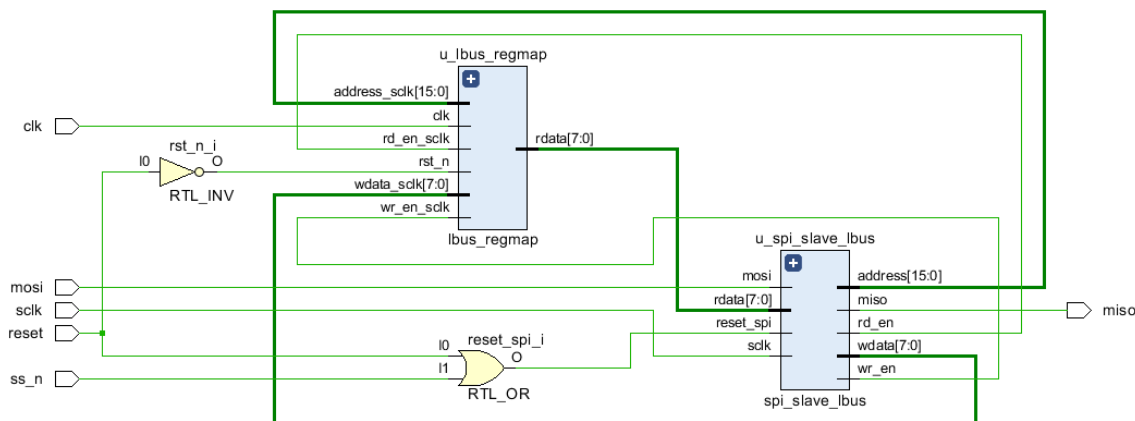


Figura 28. RTL protocolo de comunicación SPI. (Xilinx Inc., 2016).

El mapa de memoria del protocolo se modifica y redimensiona para almacenar los eventos provenientes de los ADC. La Tabla 1 muestra como se distribuyen los datos transferidos y la posición de memoria asignada para cada uno, obteniendo una totalidad de 98 posiciones con un ancho de 8 bits para cada una.

Posición Inicial	Posición Final	Dato	Bits usados	Descripción
0	1	Trigger 1	12	Parámetros ingresados por usuario
2	3	Trigger 2	12	
4	5	Trigger 3	12	
6	7	Rampa 1	12	
8	9	Rampa 2	12	
10	11	Rampa 3	12	
12	12	Pfifo_status	1	Señal de control
13	17	Muestra 1	38	Trama de datos evento
18	22	Muestra 2	38	
23	27	Muestra 3	38	
28	32	Muestra 4	38	
33	37	Muestra 5	38	
38	42	Muestra 6	38	
43	47	Muestra 7	38	
48	52	Muestra 8	38	
53	57	Muestra 9	38	
58	62	Muestra 10	38	
63	67	Muestra 11	38	
68	72	Muestra 12	38	
73	77	Muestra 13	38	
78	82	Muestra 14	38	
83	87	Muestra 15	38	
88	92	Contador 1	38	
93	97	Contador 2	38	

Tabla 1. Mapa de memoria protocolo SPI.

3.3. Hardware externo de emulación

Teniendo en cuenta que este proyecto se compone sólo de la parte digital del sistema de adquisición de LAGO, los datos de eventos que pueden ser considerados rayos cósmicos fueron simulados desde un archivo de salida ya existente. Estos fueron almacenados en memorias ROOM de una FPGA Nexys 4-DDR (nombrada de ahora en adelante solo Hardware externo) para simular el ADC en condiciones de detección. Con esta estrategia se consigue simular la electrónica de digitalización.

El Hardware externo recibe una señal de reloj a 50 MHz proveniente de una tarjeta con FPGA Icoboard para sincronizar la de información almacenada en las memorias.

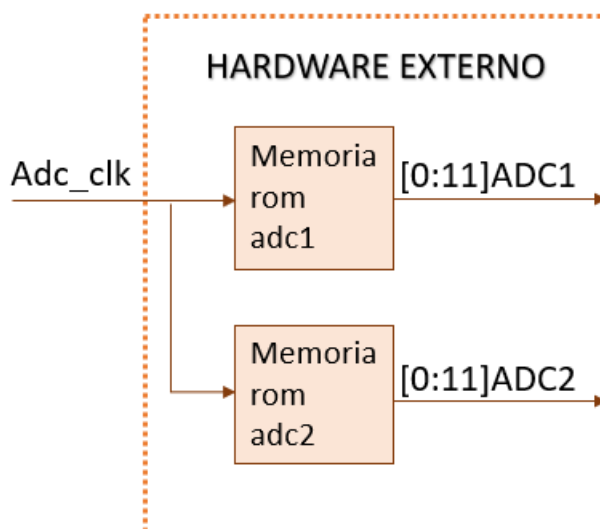


Figura 29. Diagrama de bloques hardware externo, *Adc_clk* es la señal de reloj transmitida por la Icoboard.

Ha sido necesario incluir una descripción que permite sincronizar en frecuencia y fase los relojes de las dos FPGA y así asegurar que las muestras se entreguen sin pérdida de información por violación de tiempos.

3.4. Raspberry pi 3 B+

La Raspberry almacena las tramas de datos que componen un evento y registra los datos obtenidos de tres dispositivos que conforman el sistema de adquisición a través de diferentes protocolos de comunicación como se muestra en la Figura 30.

- Protocolo SPI para FPGA Icoboard: permitiendo acceder a los registros para configurar parámetros de adquisición y envío de los eventos registrados.
- Protocolo I2C para el sensor HP03S: por medio de la librería wiringPiI2C se configura la comunicación. (Hoperf, 2020).

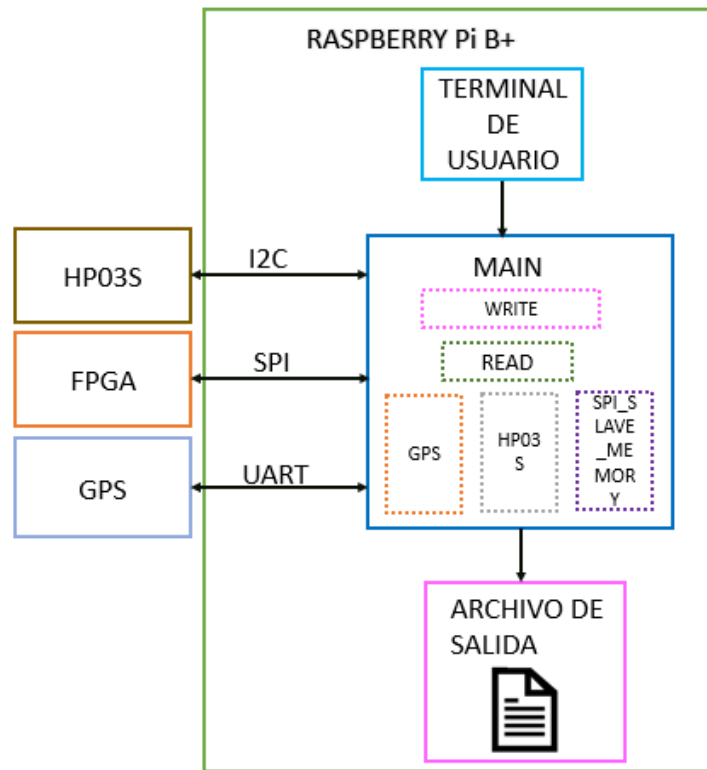


Figura 30. Diagrama de bloques Raspberry Pi.

- Protocolo UART con sensor GPS: se obtiene lectura hora y geoposicionamiento. (Adafruit, 2020).

Finalmente, el archivo Main corresponde a un *script* en Python donde se establece el llamado de todas las funciones, para recopilar toda la información. Además, se crea una interfaz de usuario para el ingreso de parámetros por medio de una terminal. Posteriormente se entrega un archivo de

salida que presenta el registro de los eventos.

4. Implementación proyecto LAGO

4.1. FPGA

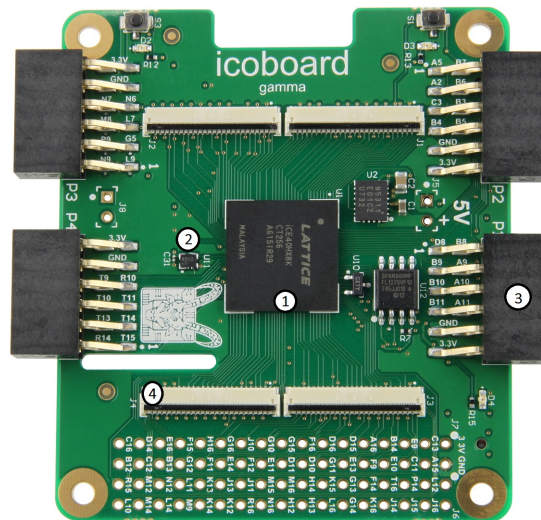


Figura 31. Icoboard con FPGA Lattice iCE40 de Trenz Electronic con SRAM de 8 MBit. (Icoboard, 2020).

En este trabajo se selecciona la placa ICOBOARD que se muestra en la Figura 31, contiene una FPGA Lattice con 8K LUT (1), un reloj de 100 MHz (2), 8 MBit de SRAM programable en Verilog por una cadena de herramientas de código abierto, 4 conectores Pmod de 16 entradas y salidas (E/S) (3) y 4 conectores Flat flex cada uno con 36 E/S (4). Es responsable del pre-procesamiento de los datos, el ajuste de umbrales de activación, el ajuste de los voltajes de compensación de línea base y regulación de la fuente de alto voltaje para los PMT.

La Icoboard es compatible con la Raspberry y es por medio de esta que se carga el archivo

ejecutable en la FPGA. La Figura 32 muestra un diagrama de bloques para la comunicación entre las dos tarjetas.

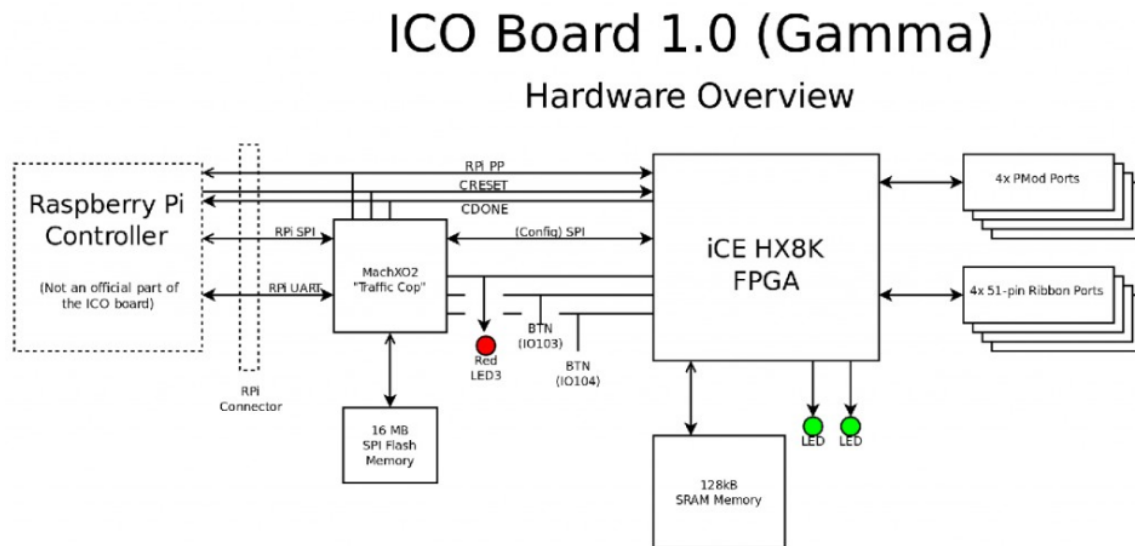


Figura 32. Compatibilidad con la Raspberry Pi. (Icoboard, 2020)

A continuación se describe de manera detallada los recursos lógicos disponibles en la FPGA Icoboard. La Figura 33 muestra su arquitectura. La FPGA tiene la disponibilidad de 4 bancos de puertos I/O los cuales son usados para la adquisición en paralelo de los 3 canales con una resolución de 12 bits, demandando una totalidad de 36 entradas.

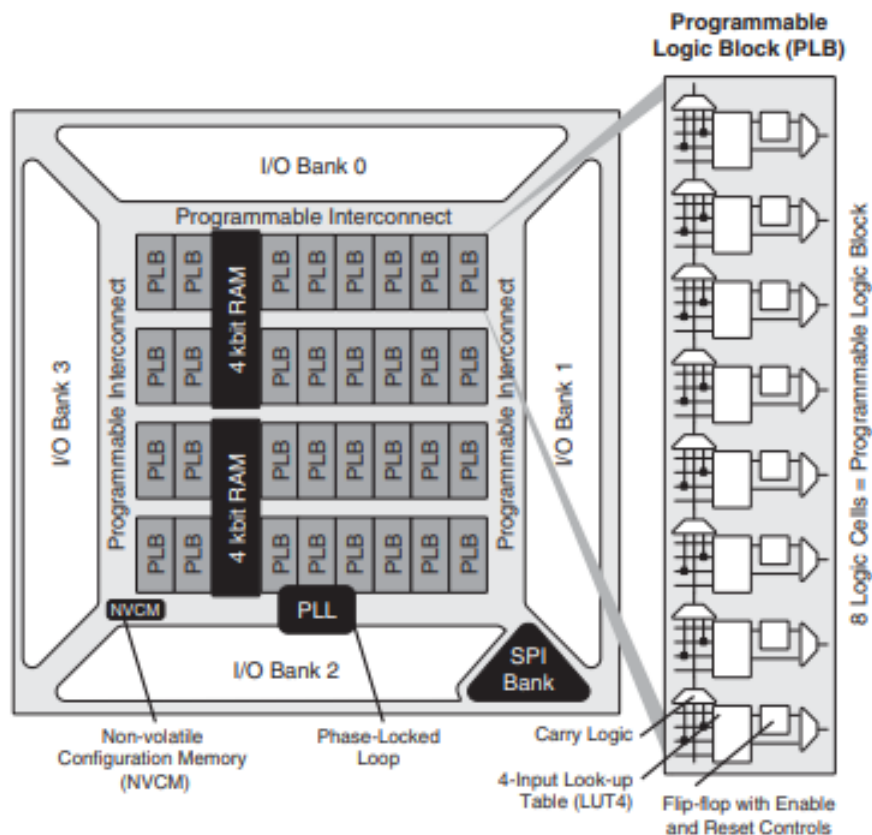


Figura 33. Arquitectura iCE40. (Lattice Semiconductor, 2017).

En la Figura 34 se muestran los recursos lógicos disponibles para la implementación de la descripción en hardware. Según los reportes de utilización que entrega el software utilizado Arachne, la implementación del proyecto necesita:

- 38 entradas/salidas
- Celdas lógicas utilizadas: 4335
- Flip Flop tipo D: 2435

Con esto se determina que la descripción en hardware necesario puede ser soportado por la FPGA seleccionada.

Part Number	LP384	LP640	LP1K	LP4K	LP8K	HX1K	HX4K	HX8K
Logic Cells (LUT + Flip-Flop)	384	640	1,280	3,520	7,680	1,280	3,520	7,680
RAM4K Memory Blocks	0	8	16	20	32	16	20	32
RAM4K RAM bits	0	32K	64K	80K	128K	64K	80K	128K
Phase-Locked Loops (PLLs)	0	0	1 ¹	2 ²	2 ²	1 ¹	2	2
Maximum Programmable I/O Pins	63	25	95	167	178	95	95	206
Maximum Differential Input Pairs	8	3	12	20	23	11	12	26
High Current LED Drivers	0	3	3	0	0	0	0	0

Figura 34. Recursos lógicos de la FPGA. (Lattice Semiconductor, 2017).

4.2. Sensores adicionales

En esta sección se describen los sensores usados para el proyecto, los cuales son de vital importancia para los análisis posterior del archivo de salida. El sensor de temperatura y presión barométrica permite conocer las condiciones atmosféricas básicas bajo las cuales trabaja el WCD, y de esta manera, se obtiene información de las variaciones de temperatura que afectan el comportamiento del detector y por tanto generar alteraciones en la tasa de eventos registrados.

Por otra parte, el GPS seleccionado (Adafruit Ultimate) permite conocer las condiciones de geoposicionamiento del detector. Además permite la sincronización temporal de la interfaz digital al usar la señal PPS (Pulse-Per-Second).

A continuación se muestra las características más relevantes de los sensores usados.

4.2.1. GPS. Un criterio importante para la configuración entre Raspberry-GPS-FPGA es la necesidad de generar un archivo de salida de datos como se plantea en los objetivos que contenga una referencia temporal precisa, para relizar estudios dinámicos. También, la información de geoposicionamiento del flujo de rayos cósmicos secundarios es un factor relevante que afecta el flujo de partículas detectadas con parámetros como la altura a nivel del mar y la latitud.

Es importante que el GPS seleccionado tenga la señal PPS (Pulse-Per-Second) como uno de sus periféricos. Esta señal y el tiempo UTC del GPS proporcionan la información de tiempo a la Raspberry Pi.

Dentro las características más relevantes del GPS está el periférico PPS, Y una interfaz de comunicación UART a 9600 baud.

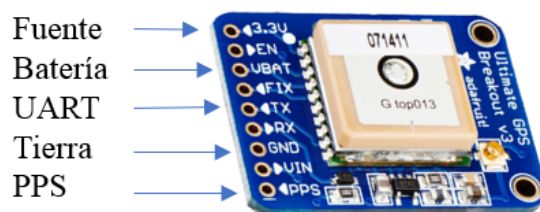


Figura 35. GPS adafruit ultimate. (Adafruit, 2020).

4.2.2. Temperatura y presión atmosférica. El sensor de temperatura y presión utilizado cuenta con una interfaz de comunicación I2C, un convertidor análogo digital con 16 bits de resolución que proporciona datos de la presión y la temperatura. El sensor es calibrado por el fabricante almacenando 11 coeficientes únicos en el chip, los cuales se usan en la lectura de presión y temperatura. Este dispositivo opera con 3V, una corriente de 500uA durante la conversión y 1uA en reposo.



Figura 36. Sensor de presión y temperatura HP03S. (Hoperf, 2020)

Parameter	Symbol	Conditions	Min	Typ	Max	Unit
Absolute Pressure Accuracy	HP03SA	750-1100 @ 0~50	-1.5		1.5	hpa
Absolute Pressure Accuracy	HP03SB	750-1100 @ 0~50	-3.0		3.0	hpa
Absolute Pressure Accuracy	HP03SA	750-1100 @ -20~60	-3.0		3.0	hpa
Absolute Pressure Accuracy	HP03SB	750-1100 @ -20~60	-5.0		5.0	hpa
Long Term Stability		12 month		2		hpa
VDD Dependency		2.4~3.6	-1.5	0	1.5	hpa
Temperature Accuracy		0~+50	-1.0		1.0	°C
Temperature Accuracy		-20~+60	-2.0		2.0	°C

Figura 37. Características de salida del sensor de presión y temperatura. (Hoperf, 2020)

4.3. Raspberry Pi B+

Se eligió la Raspberry Pi3 Model B+ puesto que esta cumple con los requerimientos necesarios para el desarrollo del proyecto.

Procesador	Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC
Frecuencia de reloj	1,4 GHz
Memoria	1GB LPDDR2 SDRAM
Conectividad inalámbrica	2.4GHz / 5GHz IEEE 802.11.b/g/n/ac Bluetooth 4.2, BLE
Conectividad de red	Gigabit Ethernet over USB 2.0 (300 Mbps de máximo teórico)
Puertos	GPIO 40 pines, HDMI, 4 x USB 2.0, CSI (cámara Raspberry Pi), DSI (pantalla tácil), Toma auriculares / vídeo compuesto, Micro SD, Micro USB (alimentación), Power-over-Ethernet (PoE)

Tabla 2. Características Raspberry Pi3 Model B+. (Pasor, 2018)

Sus características más destacadas se muestran en la Tabla 2. Tiene una frecuencia de 1,4 GHz para sus cuatro núcleos ayudando a obtener mejores rendimientos en muchas de las tareas que se

propongan a este miniPC. La plataforma soporta diversos sistemas operativos, desde GNU/Linux hasta algunas versiones de Windows, para el caso del proyecto se utiliza Raspian (distribución de GNU/Linux basado en Debian para Raspberry PI).

Uno de los retos que se presentan en el desarrollo del proyecto es adquirir los datos de interés y almacenarlos, por esta razón se utilizaron librerías que permiten el uso de pines GPIO (General Purpose Input Output) con el lenguaje de medio nivel (C o C++) y Python, esto con el fin de hacer uso eficiente de los recursos del sistema.

4.4. Integración de periféricos

Este circuito se implementa para tener la conexión de la Raspberry y la FPGA con los periféricos de una manera más organizada y así lograr la comunicación de todos los componentes utilizados en el proyecto.

La PCB se diseña a doble capa, esta es compatible con una amplia gama de Raspberry y tiene dimensiones de 56.4 x 65.9 mm. Ver Figura 39.

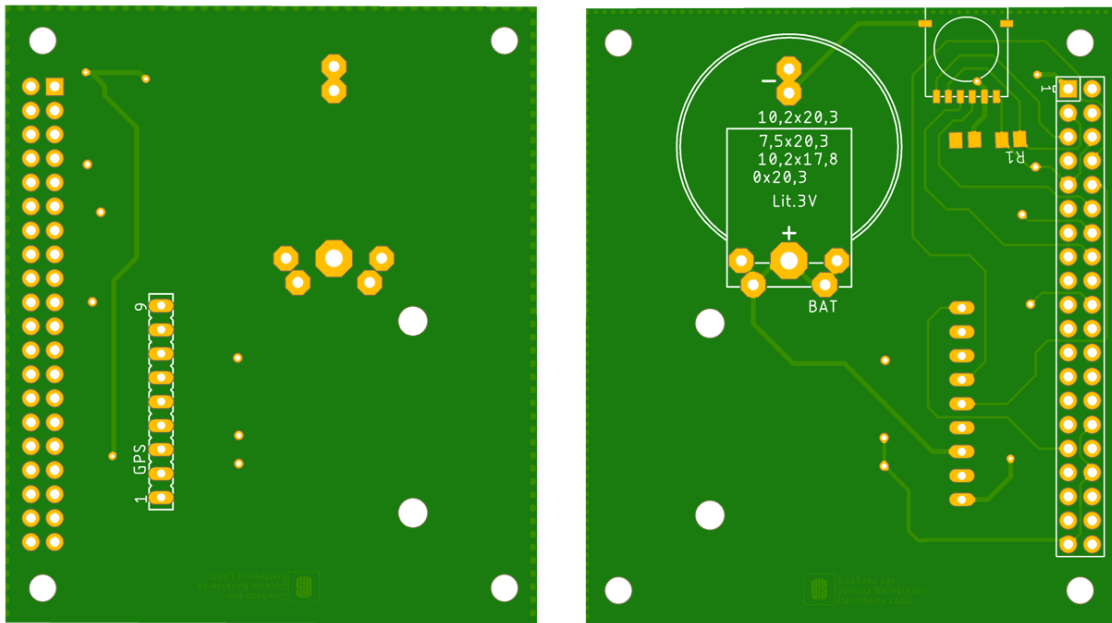


Figura 38. Diseño de la PCB

En la Figura 39 se muestra una tarjeta "shield" donde se realiza la comunicación teniendo en cuenta cada uno de los protocolos que permiten recibir la información suministrada por los sensores. En la PCB diseñada se realizaron conexiones del protocolo UART para el GPS Adafruit Ultimate e I2C para el sensor de presión y temperatura HP03S con pines de la Raspberry pi3 B+.

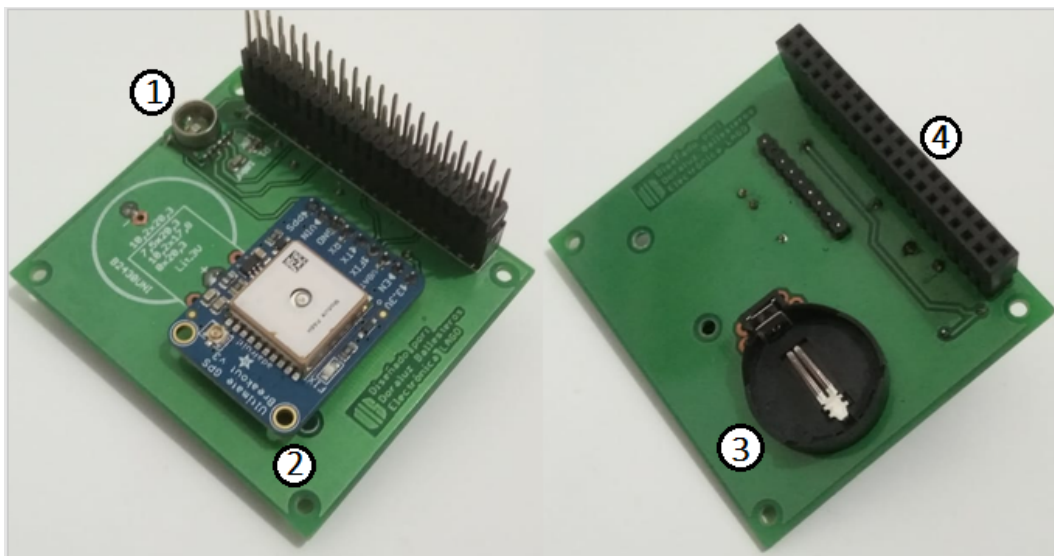


Figura 39. PCB de integración de periféricos

Se integran los periféricos captadores así: sensor HP03S (1), GPS Adafruit Ultimate (2), batería (3), conector 40 pines (4).

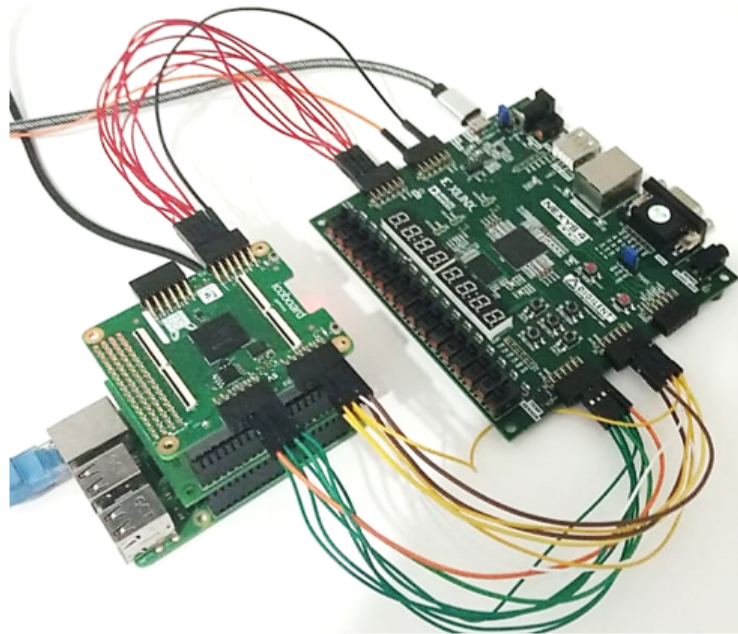


Figura 40. Implementación proyecto LAGO para pruebas con nexys 4-DDR

En la Figura 40 se puede observar la conexión implementada para simular la señales provinientes de la tarjeta digitalizadora presentada en la Figura 6 por medio de una Nexys 4-DDR.



Figura 41. Implementación del hardware LAGO actualizado

En la Figura 41 se encuentra ubicado de abajo hacia arriba la Raspberry PI model B+, PCB con periféricos y la FPGA Icoboard.

Note que la Raspberry alimenta a todo el conjunto de tarjetas, esta entrega una tensión de 3,3 V.

El diseño final del sistema de adquisición se muestra en la Figura 41, nótese que se ve mucho más compacto que el LAGO original Figura 57, excepto por nexys 4-DDR usada como hardware externo para las pruebas Figura 40.

5. Resultados

Para verificar el correcto funcionamiento de los circuitos diseñados descritos a nivel rtl en la FPGA, se toma un archivo de los existentes en LAGO y se ingresa a una FPGA de emulación en este caso la Nexys 4, estos son enviados a la FPGA Icoboard para su posterior procesamiento y para determinar los eventos que finalmente se almacenan en un archivo de salida junto con la información entregada por los periféricos.

En esta sección se evidencia en primera instancia las simulaciones a nivel RTL de los circuitos descritos en HDL para la actualización del proyecto LAGO. A su vez los resultados se comparan con las mediciones del hardware en el proyecto original evidenciando la respuesta del sistema ante una entrada conocida.

Cabe resaltar que las simulaciones que se presentan a continuación , tienen como entrada una base de datos en representación binaria a 12 bits y formato .mem, la cual se genera con eventos extraídos de los archivos de salida obtenidos con la versión original del proyecto LAGO. Dichos archivos han sido validados por el grupo de investigación Halley encargado del proyecto en Co-

lombia.

5.1. Comparación simulaciones de LAGO

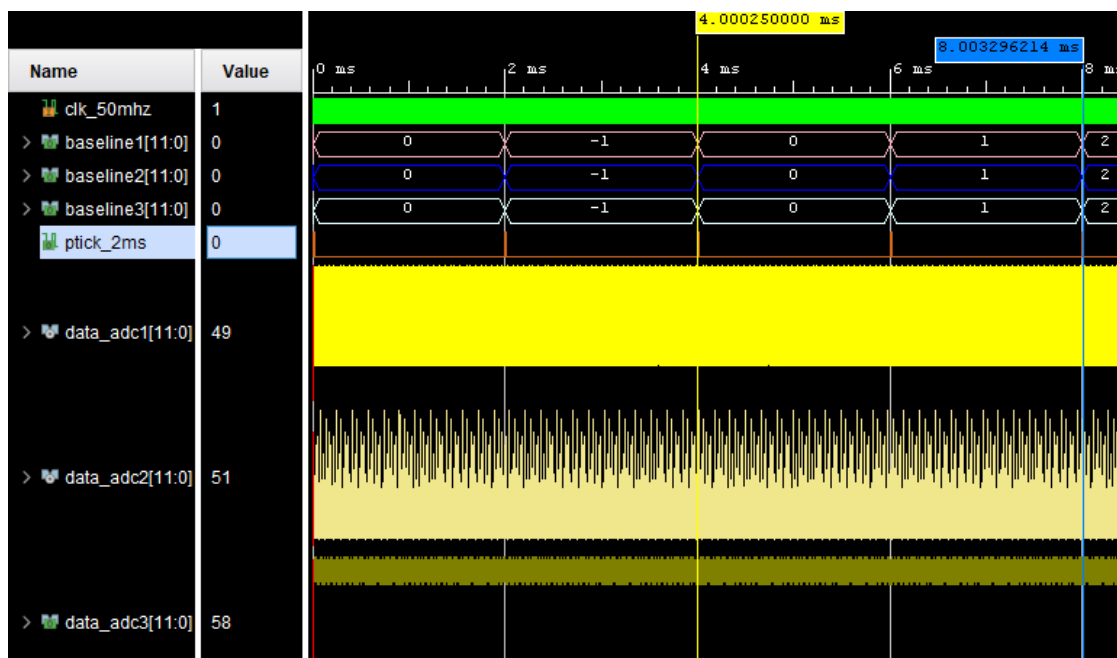


Figura 42. Simulación del comportamiento de las señales en baseline, versión actualizada.

5.1.1. Simulación Baseline versión original y versión actualizada. En la simulación mostrada en la Figura 42 en color café se muestra la actualización de la señal (ptick_2ms) con un período de 2ms.

A cada canal de entrada (data_adc1, data_adc2, data_adc3) se le ingresa una señal diferente desde la memoria del hardware externo. Las señales de salida (baseline_1, baseline_2, baseline_3), tienen el mismo comportamiento, ya que en el diseño se tiene un contador que acumula por 2 ms los datos provenientes de cada canal y calcula un promedio del error en la entrada con respecto a una referencia de 50 mV que en este caso corresponde a 50 niveles de cuantización. La salida de este circuito será conectada a un DAC para generar el voltaje necesario y así, corregir el offset de

línea base para mantener el sistema en estado estable.

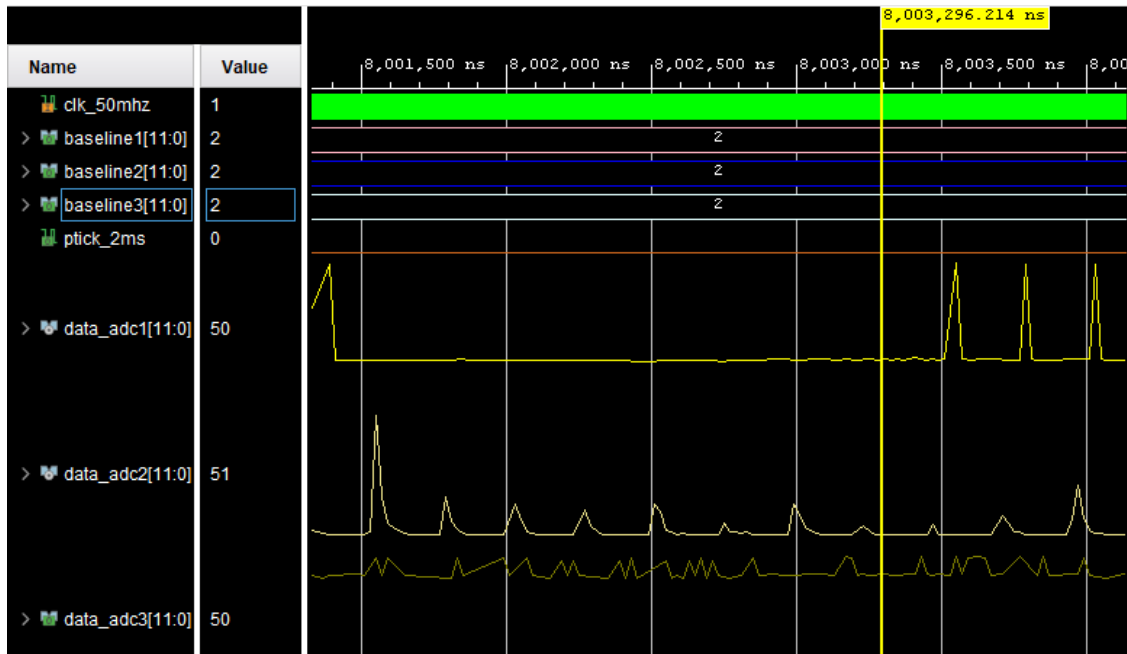


Figura 43. Simulación de las entradas de baseline, versión actualizada.

En la Figura 43 se hace un acercamiento al cursor posicionado en 8,00329996214 ms mostrado en la figura anterior, para observar el comportamiento de las entradas en forma analógica, nótese el comportamiento diferente que tienen las tres entradas.

El comportamiento en las señales de salida (baseline1, baseline2, baseline3) es similar logrando una actualización cada 2 ms.

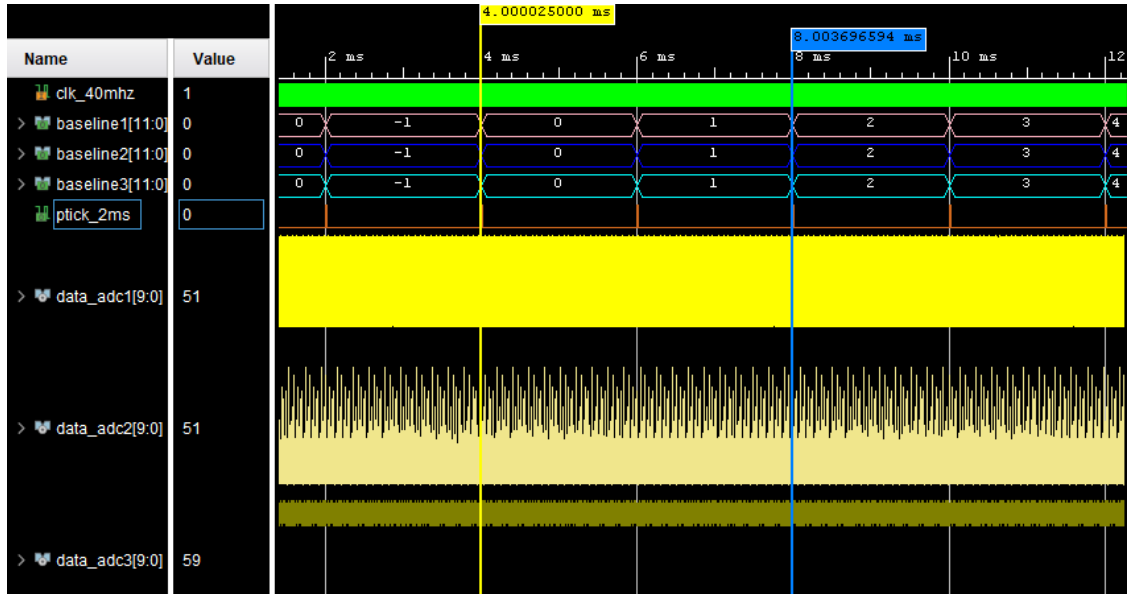


Figura 44. Simulación del comportamiento de las señales en baseline, versión original.

En la Figura 45, el cursor amarillo se ubica en el mismo instante de tiempo que en Figura 43, pero las entradas (data_adc) no se encuentran en el mismo valor numerico que se presenta en la columna (value), eso se debe a la mejora en la frecuencia de la versión actualizada, que se implementa a 50 MHz con respecto a la original implementada a 40 MHz . De acuerdo a lo anterior los datos que se observan son diferentes.

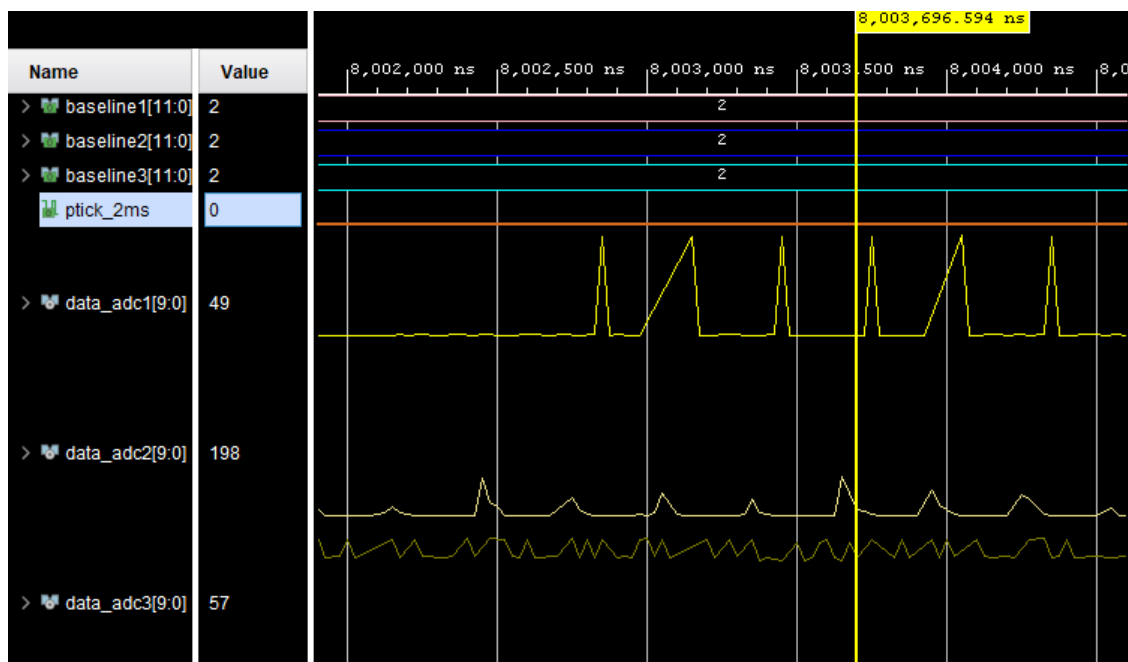


Figura 45. Simulación de las entradas de baseline, versión original.

5.1.2. Simulación Rampa versión original y versión actualizada. En esta simulación se evidencian algunos cambios realizados respecto al diseño original LAGO con el fin de aplicar unas mejoras. Una es la restricción del máximo valor que el usuario puede ingresar como parámetro de entrada DATA_IN(1023) a la rampa. Dado que en la versión original aunque el valor máximo efectivo es 1023, es posible ingresar un número mayor así no tenga efecto alguno en la generación de la señal de salida.

En la Figura 46 se puede observar las rampas que se generan a partir de tres valores de polarización que se ingresan en distintos instantes de tiempo por el usuario, con tiempo de 200 ms en estado estable para cada uno.

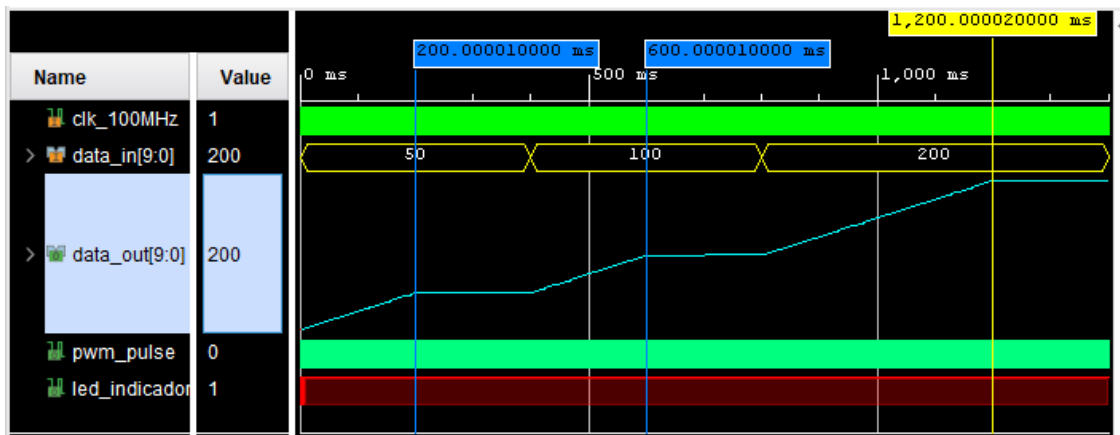


Figura 46. Simulación rampa de polarización del proyecto LAGO, versión actualizada.

En Figura 47 se observa un acercamiento donde se puede apreciar el comportamiento del ciclo útil en la señal (PWM_pulse) con duración de 20 ns.

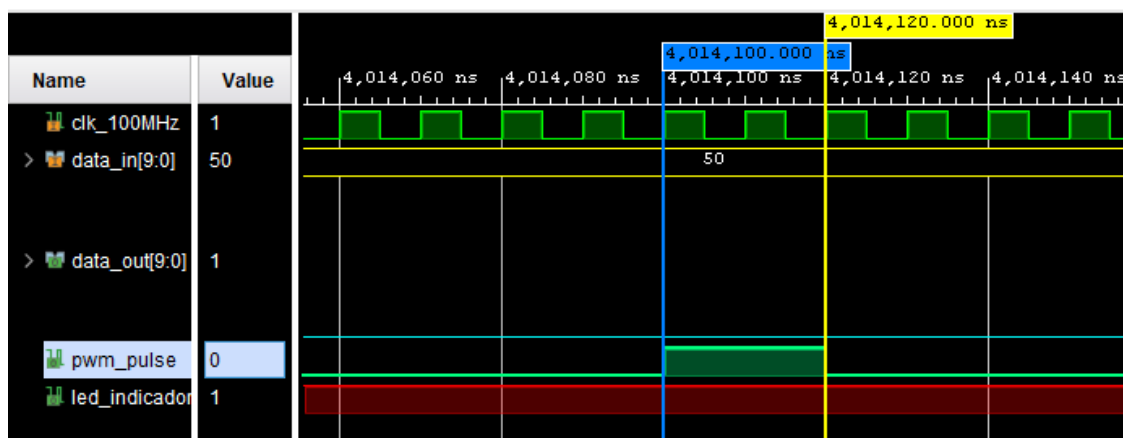


Figura 47. Ciclo útil de 10 ns para la PWM del proyecto LAGO, versión actualizada.

■ Simulación Rampa versión original

Para estimular este diseño se ingresa un DATA_IN de 1023, de igual manera se estimula con un período de 20 ns ya que la frecuencia de la FPGA utilizada es de 50 MHz. La salida (data_out) cambia cada 4 ms como se muestra en la Figura 49.

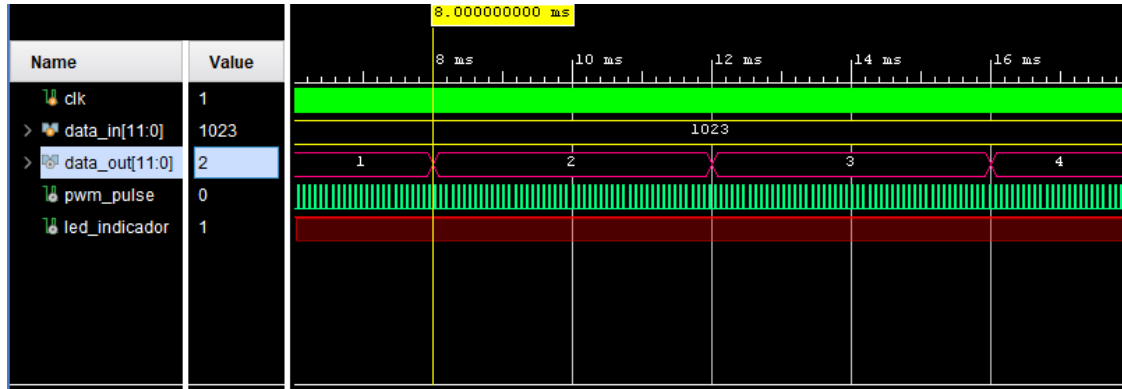


Figura 48. Actualización de la PWM del proyecto LAGO, versión original.

A diferencia del diseño original de LAGO este diseño tiene un ciclo útil de 20 ns lo cual se evidencia en logrando actualizar la PWM de manera más rápida.

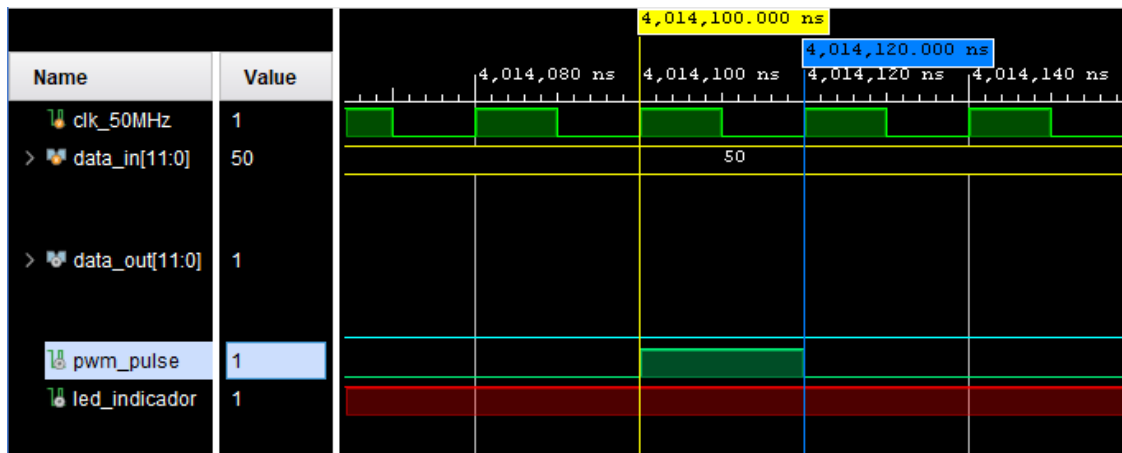


Figura 49. Ciclo útil de 20ns para la PWM del proyecto LAGO, versión original.

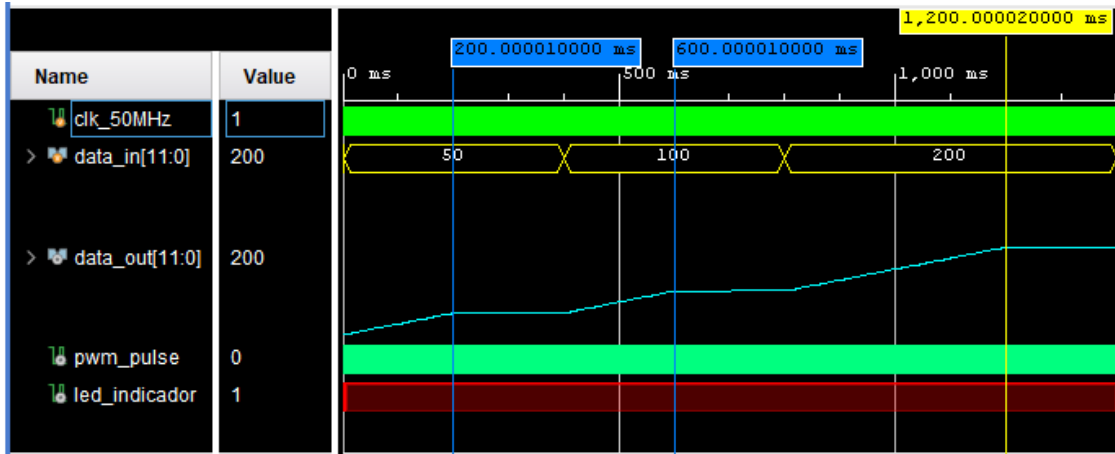


Figura 50. Ciclo útil de 20ns para la PWM del proyecto LAGO, versión original.

En la Figura 50 se estimula el circuito con las condiciones presentadas en la Figura 46.

Las simulaciones evidencian que las dos versiones presentan el mismo tiempo de establecimiento el cual se puede ver en los cursores de la Figura 50 y la Figura 46. De lo anterior podemos concluir que conserva la misma respuesta la señal de salida.

5.1.3. Simulación Trigger versión original y versión actual. Las señales se estimulan con un reloj de periodo 20 ns.

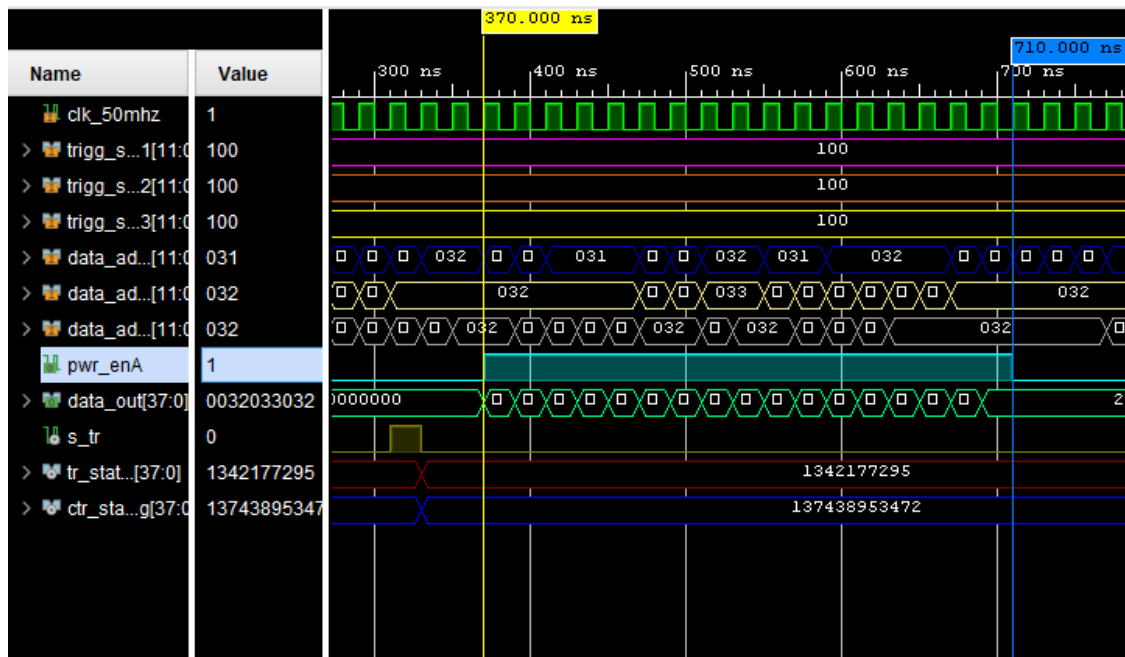


Figura 51. Señales de Trigger destacadas del proyecto LAGO, versión actualizada.

La simulación muestra el circuito en estado de detección que evidencia la secuencia de señales necesarias para el almacenamiento del evento considerado rayo cósmico. Ver Figura 51

La señal (s_tr) se activa en el momento que cualquier canal de entrada (Data_adc) supera sus respectivos umbrales (trigger_set), dos ciclos de reloj después, se pone en alto (pwr_enA) durante 340 ns indicando que se almacenarán las 15 muestras que ingresan de la captura.

La señal (data_out) concatena las muestras de los tres canales para su posterior comunicación por protocolo SPI con la Raspberry Pi, de modo que se envían consecutivamente 15 tramas de datos correspondientes al evento y dos adicionales para los contadores (tr_status_reg) y (ctr_status_reg).

■ Simulación Trigger versión original

En la Figura 52 cabe notar algunas diferencias que se presentan por las mejoras realizadas al diseño. Se estimularon las señales con un reloj de 40 MHz, Data_out se respresenta en 32 bits, el

tiempo en que pwr_enA esta en alto de 350 ns. Al igual que la figura anterior se establecen los mismo estímulos de entrada para evidenciar un comportamiento similar en los diseños.

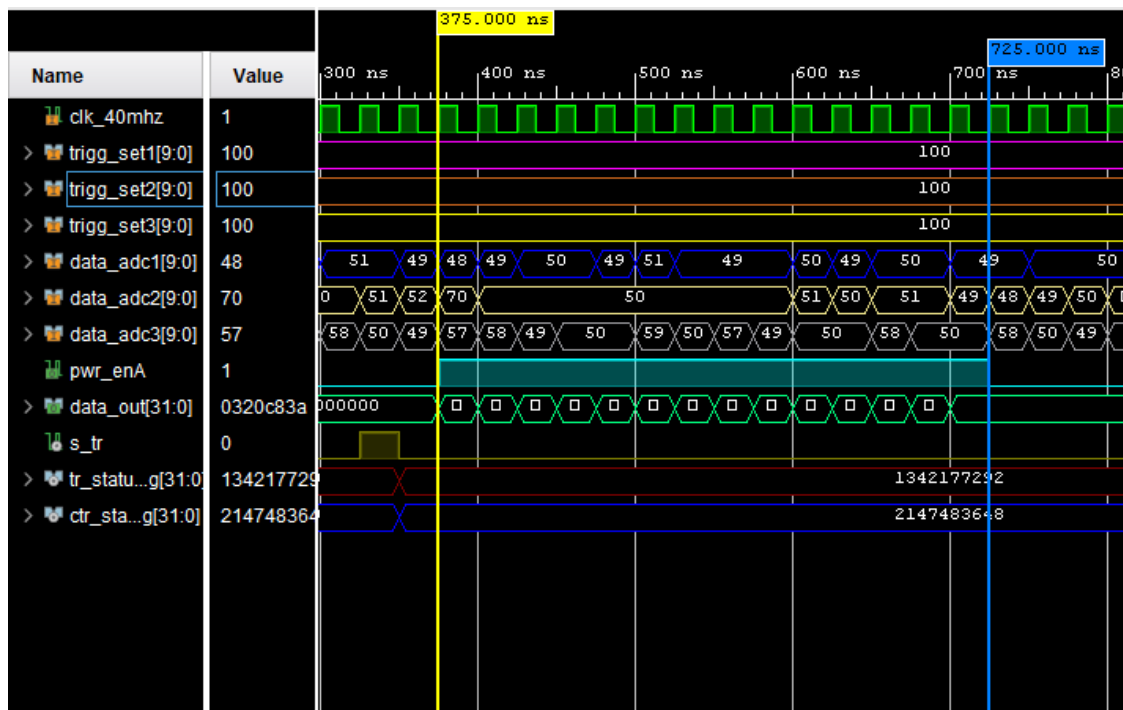


Figura 52. Señales de Trigger destacadas del proyecto LAGO, versión original.

Como se evidencia en cada una de las simulaciones se logra obtener un resultado equivalente en las señales de control que componen el procesamiento de datos del proyecto LAGO y algunas mejoras como el aumento en la frecuencia de muestreo.

5.2. Salida de control para PMT's

A continuación se muestran graficas de la señal PWM obtenida luego de implementar en FPGA los diseños expuestos en Capítulo 3.

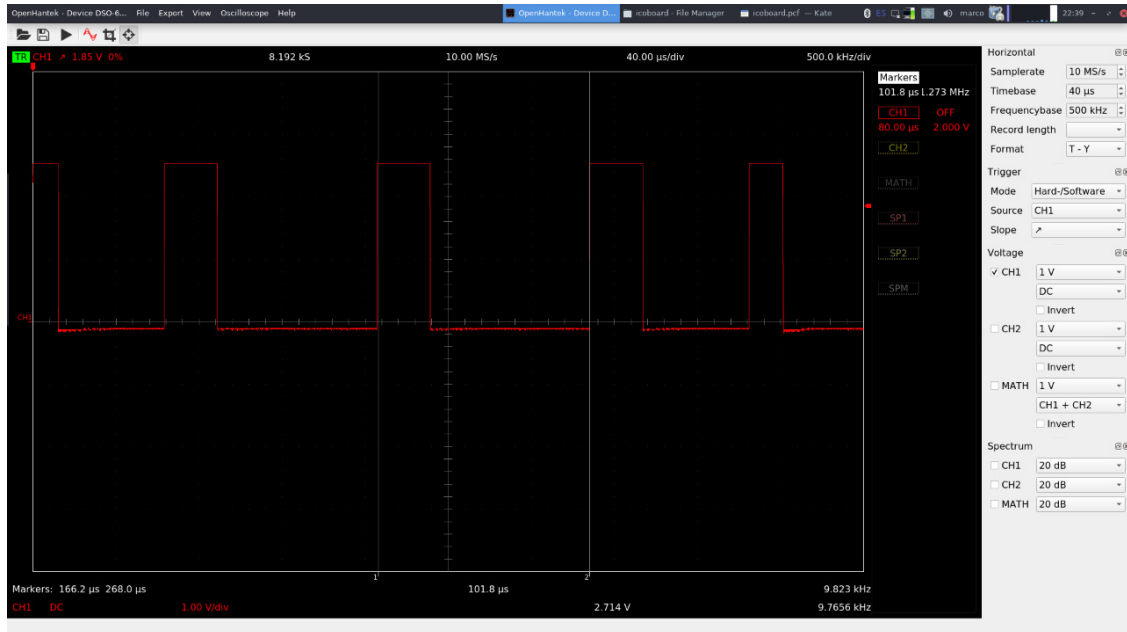


Figura 53. Señal de control PWM, versión actualizada.

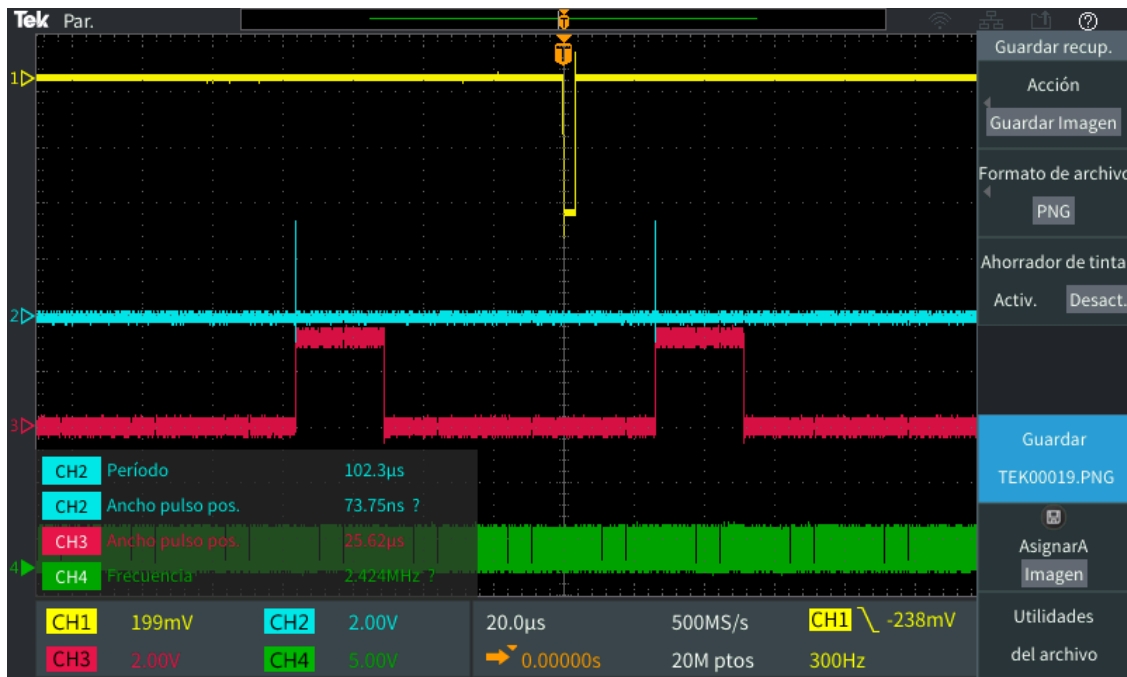


Figura 54. Señal de control PWM, versión actual.

En la Figura 53 se obtiene una señal que cumple con una frecuencia medida en osciloscopio de 9,823 KHz de ciclo útil variable y un voltaje máximo de 2,714 V.

En la Figura 54 se observa la señal PWM en el CH2 con un período de 102,3 us equivalente a una frecuencia de 9,775 KHz con un ciclo útil variable.

5.3. Registro de datos

En primer lugar se resalta la terminal de usuario Figura 57 en la que se ingresan cada uno de los umbrales ya explicados durante el desarrollo del proyecto.

```

2020-04-04 23:56:23.944305
v 1

This is a LAGO raw data file, version 1
It contains the following data:
# <N1> <N2> <N3> : line with values of the 3 ADC for a triggered pulse
# t <C> <V>, : end of a trigger
gives the channel trigger (<C>: 3 bit mask) & 50MHZ clock count (<V>) of the trigger time
# c <C> : internal trigger counter
# x f <V> : 50 MHz frequency
# x r C1-DD <V> : raw temperature & pressure sensor value
# x r D1 <V> : raw temperature/pressure value
# x r D2 <V> : raw temperature/pressure value
# x h <HH:MM:SS> <DD/MM/YYYY> <S> : GPS time (every new second, last number is seconds since EPOCH)
# x s <T> C <P> hPa <A> m : temperature <T>, pressure <P> & altitude (from pressure) <A>
# x g <LAT> <LON> <ALT> : GPS data - latitude, longitude, altitude
In case of error, an unfinished line will be finished by # E @@@
Followed by a line with # E <N> & the error message in human readable format, where <N> is the error code:
# E 1 : read timeout of 2 seconds
# E 2 : too many buffer reading tries
# E 3 : unknown word from FPGA

Current registers setting

Ingrese cada uno de los umbrales teniendo en cuenta que por no puede ser mayor a 1023
Umbral T1:
# x T1,1000
1000
Umbral T2:
# x T2,100
100
Umbral T3:
# x T3,1000
1000
Umbral HV1:
# x HV1,100
100
Umbral HV2:
# x HV2,600
600
Umbral HV3:
Temp : 38.71°C
Pressure at sealevel: 974.94hPa
Pressure : 900.60hPa@664.0m
Altitude estimated : 983.18m
# x HV3,100
100

```

Figura 55. Cabecera del archivo de datos del terminal de usuario proyecto LAGO.

Luego de la implementación se obtiene el siguiente archivo de salida en el cual se evidencia la correcta comunicación entre la Icoboard y la Raspberry Pi para registrar el evento proveniente de

la FPGA Nexys 4 a una frecuencia de 50 MHz.

En la Figura 56 se muestra en recuadro rojo un recorte de uno de los eventos obtenidos en el archivo de salida como resultado de este proyecto, y en recuadro azul un recorte de uno de los eventos tomado como referencia del proyecto LAGO.

# t 2 128271154	# t 2 809695
# c 2147535245	# c 177517050
50 51 1	178 51 50
50 56 1	50 56 49
50 111 1	50 111 58
49 64 1	51 64 51
50 61 1	50 61 50
50 48 1	50 48 50
50 51 1	51 51 59
50 51 1	51 51 50
58 51 1	50 50 50
50 63 1	50 49 50
50 50 1	50 50 59
50 50 1	50 50 50
50 63 1	# t 2 1030232
59 50 1	# c 177517051
58 50 1	177 48 50
# t 2 128288654	50 72 50
# c 2147535246	50 283 50
50 49 1	49 107 50
51 72 1	51 115 59
59 283 1	50 58 58
50 107 1	50 53 50
51 115 1	49 52 50
59 58 1	50 52 50
50 53 1	49 52 50
50 52 1	51 51 59
51 52 1	49 50 58
51 52 1	# t 2 1252175
58 51 1	# c 177517052
50 50 1	178 49 50
50 51 1	50 52 50
50 50 1	50 278 58
50 49 1	50 133 51
# t 2 128323656	50 72 50
# c 2147535248	50 66 50
	51 51 51

Figura 56. Comparación de archivos de salida con datos de eventos considerados rayos cósmicos. *Cuadro rojo: versión actualizada. *Cuadro azul: versión original.

En la Figura 57 se muestra un evento para corroborar el funcionamiento del diseño implementado y mostrar el comportamiento de los datos obtenidos, además, una interpolación de las muestras registradas usando Matlab ® para tener un acercamiento real de la forma del pulso de un evento considerado rayo cósmico.

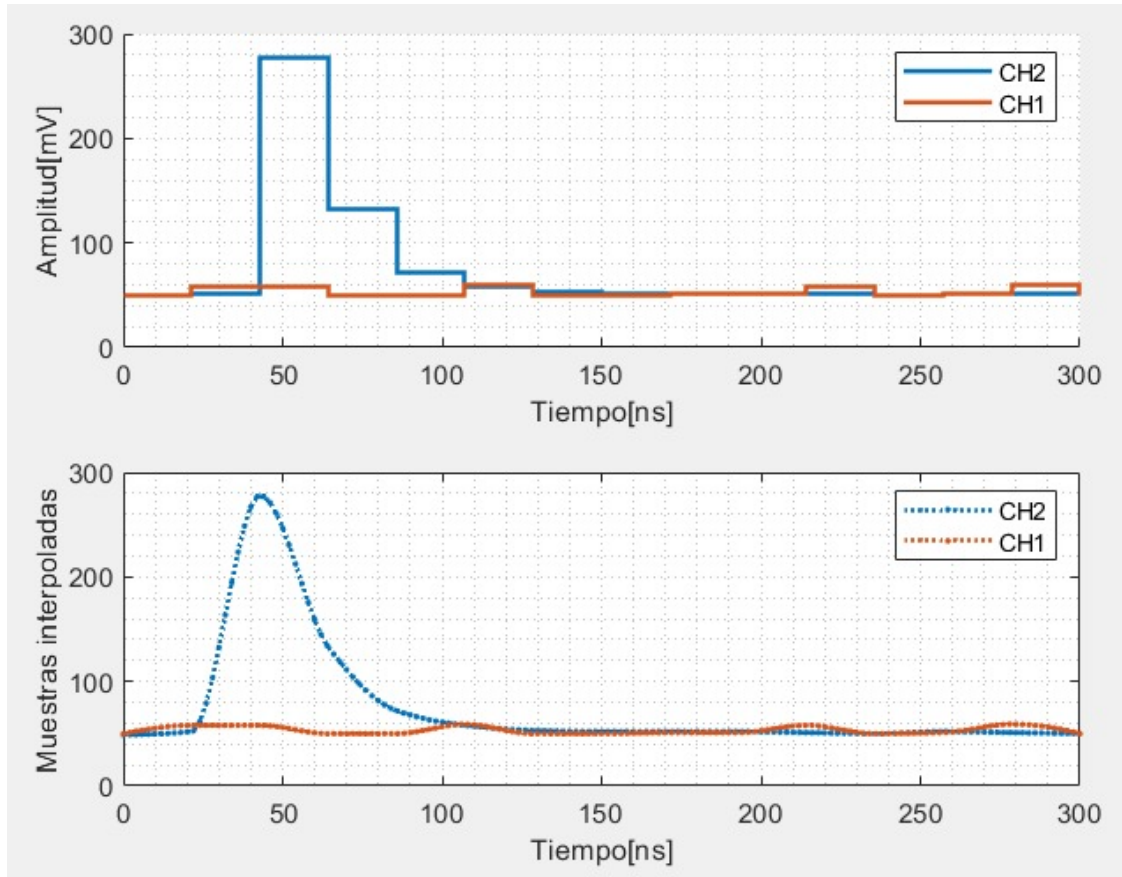


Figura 57. (Arriba) Gráfica discreta de un evento registrado por el hardware diseñado.(Abajo) pulso interpolado.

En las siguientes gráficas se resume el comportamiento de los datos entregados por el sensor de presión y temperatura, estos datos son muy importantes en el proyecto LAGO ya que mediante el análisis de estos se logra hacer calibraciones en los WCD.

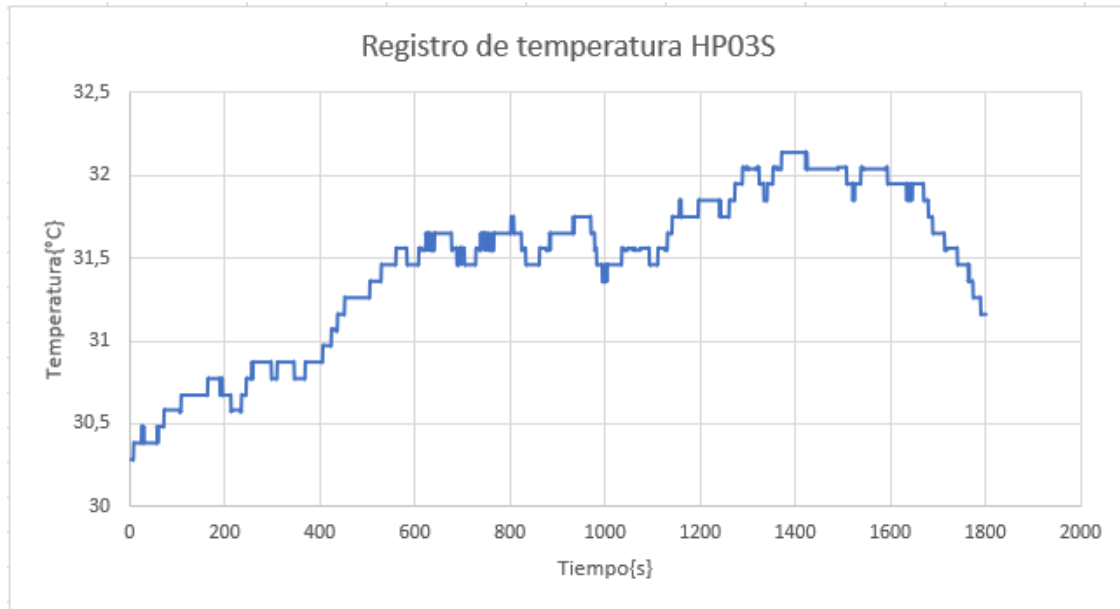


Figura 58. Registro de temperatura HP03S durante un registro de media hora.

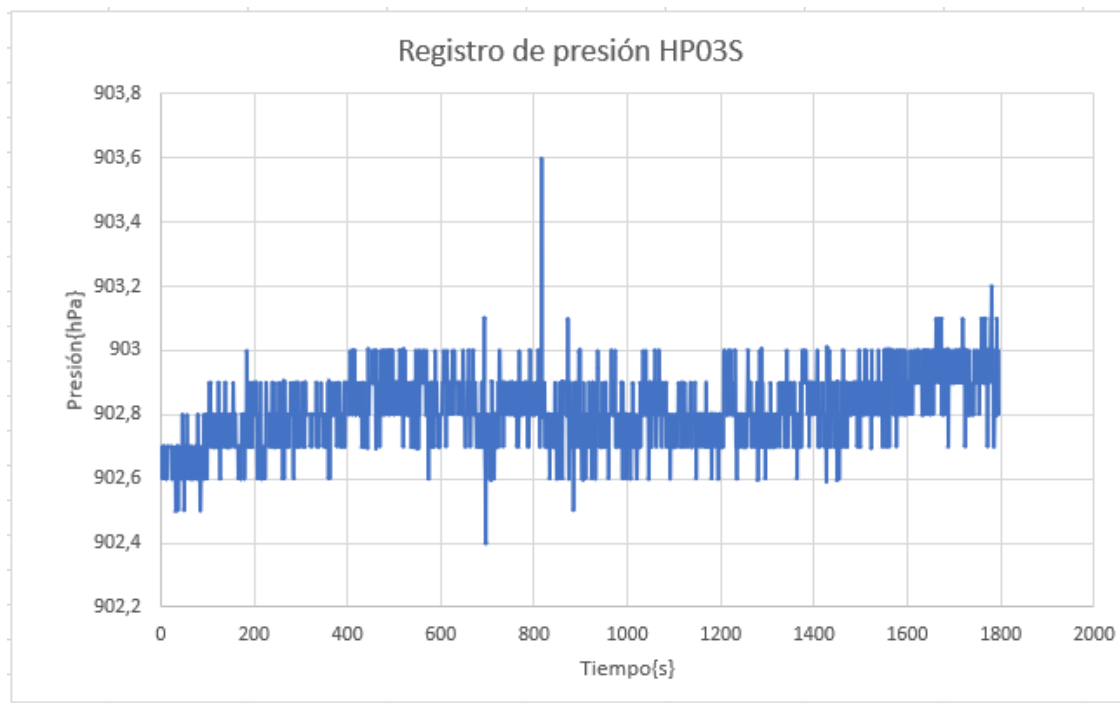


Figura 59. Registro de presión HP03S durante un tiempo de registro de media hora.

Finalmente con los resultados mostrados se corrobora el correcto funcionamiento del diseño implementado, se logra mostrar el comportamiento de los datos que conforman un evento. A su

vez se logra con este diseño se logra mejorar la resolución temporal y la resolución vertical. Esta mejora permite disminuir el error en la estimación de la dirección de la lluvia, a su vez se logra disminuir el error en la energía depositada en los WCD y a partir de esto determinar cual partícula primaria desencadenó la lluvia.

6. Trabajo futuro

Para dar por terminada la actualización del sistema de adquisición del proyecto LAGO se debe aplicar una mejora a la electrónica analógica, aumentando la frecuencia de muestreo y la resolución para adaptarlo a la implementación de este proyecto.

7. Conclusiones

En este proyecto se diseñó un sistema de preprocesamiento digital para el registro de aquellos eventos que pueden ser considerados rayos cósmicos, se implementaron circuitos a nivel RTL en una FPGA con herramientas libres, esta versión obedece a una actualización del proyecto LAGO mejorando la frecuencia de muestreo del evento captado de 40 MHz a 50 MHz y conservando el comportamiento en las señales de control.

Se almacena la información de los eventos registrados en una Raspberry Pi3 model B+ comunicada con la FPGA Icoboard por medio de protocolo SPI, además se registró los datos de los periféricos HP03S y GPS que permiten obtener información relacionada con el entorno del WCD

(ubicación geográfica, hora, temperatura ambiente y presión atmosférica).

Referencias

- Adafruit (2020). Adafruit Ultimate GPS v3. [urlhttps://learn.adafruit.com/adafruit-ultimate-gps/](https://learn.adafruit.com/adafruit-ultimate-gps/).
- Asorey, H. (2012). *Los detectores Cherenkov del observatorio Pierre Auger y su aplicación al estudio de fondos de radiación*. Universidad Nacional de Cuyo, Comisión Nacional de Energía Atómica, Argentina.
- Charkster (2020). [urlhttps://github.com/charkster/spi_slave_verilog](https://github.com/charkster/spi_slave_verilog).
- Estupiñán López, A., Nunez, L., and Asorey, H. (2015). *Metodo de Thinning y Dethinning para lluvias de primarios con alta energía*. PhD thesis.
- Haro, M. S., Arnaldi, L. H., Alvarez, W., Alvarez, C., Araujo, C., Areso, O., Arnaldi, H., Asorey, H., Audelo, M., Barros, H., et al. (2016). The data acquisition system of the latin american giant observatory (IAGO). *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 820:34–39.
- Hernandez Barajas, S. P., Carreño, L., and Fabian, Y. (2018). *Procedimiento De Instalacion, Calibracion Y Sincronizacion Del Arreglo De Detectores Cherenkov De Agua (Guane), Para La Deteccion Y Estudio De Rayos Cosmicos En Bucaramanga*. PhD thesis, Universidad Industrial de Santander, Escuela de Ingeniería Eléctrica . . .
- Hoperf (2020). HP03S datasheet. [urlhttps://datasheetspdf.com/pdf-file/748092/HOPERF/HP03S/1](https://datasheetspdf.com/pdf-file/748092/HOPERF/HP03S/1).
- Icoboard (2020). icoBoard. [urlhttp://icoboard.org/](http://icoboard.org/).

- Kaptanoglu, T. (2018). Characterization of the hamamatsu 8” r5912-mod photomultiplier tube. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 889:69–77.
- Lattice Semiconductor (2017). DS1040 - iCE40 LP/HX Family Data Sheet. (October).
- Pasor, J. (2018). Raspberry Pi 3 B+, análisis: review con características, precio y especificaciones. [urlhttps://www.xataka.com/ordenadores/raspberry-pi-3-model-b-analisis-mas-potencia-y-mejor-wifi-para-un-minipc-que-sigue-asombrando](https://www.xataka.com/ordenadores/raspberry-pi-3-model-b-analisis-mas-potencia-y-mejor-wifi-para-un-minipc-que-sigue-asombrando).
- Schlaepfer, H. (2003). Cosmic rays. in: Spatium. *International Space Science Institute*, 40:418–462.
- Supanitsky, A. D. (2007). *Detectores de Superficie y la Composición Química de los Rayos Cósmicos*. PhD thesis, Universidad de Buenos Aires.
- Xilinx Inc. (2016). Integrated Logic Analyzer v6.2 - LogiCORE IP Product Guide. page 31.